

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Кваліфікаційна наукова
праця на правах рукопису

Коровій Олександр Сергійович

УДК 004.91

ДИСЕРТАЦІЯ
МОДЕЛІ ТА МЕТОД РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ
ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ В УНІВЕРСАЛЬНИХ
КОМП'ЮТЕРНИХ СИСТЕМАХ

123 – Комп'ютерна інженерія

12 – Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ О.С. Коровій

Науковий керівник Терейковський Ігор Анатолійович, доктор технічних наук, професор

Київ – 2026

АНОТАЦІЯ

Коровій О.С. Моделі та метод розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії з галузі знань 12 Інформаційні технології за спеціальністю 123 Комп'ютерна інженерія. – Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, 2026.

Дисертаційна робота присвячена вирішенню *наукового завдання* яке полягає у розробці нейромережевих моделей та методу, що забезпечують підвищення ефективності розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах в умовах обмежених обчислювальних ресурсів з урахуванням лінгвістичної специфіки української мови. Актуальність цього наукового завдання зумовлена безпрецедентним зростанням обсягів неструктурованих текстових даних у соціальних мережах, медіа та на онлайн-платформах, що формує високий попит на технології автоматичного розпізнавання емоційної тональності з боку державних органів та бізнес-структур. Для українського інформаційного простору ця проблематика набуває стратегічного значення в умовах повномасштабного збройного вторгнення, коли інформаційно-психологічні операції противника цілеспрямовано використовують емоційну маніпуляцію через соціальні мережі для дестабілізації суспільних настроїв і підриву стійкості держави. Разом із тим більшість існуючих засобів розпізнавання емоційної тональності потребує суттєвої адаптації до лінгвістичної специфіки української мови, характеризується високими вимогами до обчислювальних ресурсів і залежністю від закордонної інфраструктури, що обмежує їх надійність та достовірність результатів. Це визначає доцільність розробки нейромережевих моделей і методу, орієнтованих на ефективне розгортання у вітчизняних інформаційних системах за умов обмежених обчислювальних ресурсів.

Об'єктом дослідження дисертаційної роботи є процеси розпізнавання емоційної тональності текстових фрагментів в універсальних комп'ютерних системах.

Предметом дослідження є моделі та метод розпізнавання емоційної тональності текстових фрагментів.

Метою дисертаційного дослідження є підвищення ефективності засобів для розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах за рахунок підвищення якості розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

Відповідно до мети дослідження сформульовано такі наукові завдання:

1. Провести аналіз сучасних методів і засобів розпізнавання емоційної тональності тексту та визначити перспективні шляхи підвищення їх ефективності.

2. Провести розвиток методологічної бази розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах.

3. Розробити метод розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах.

4. Розробити та дослідити систему розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах.

Основні наукові результати дисертаційного дослідження:

- *Вперше* розроблено модель побудови нейромережових засобів розпізнавання емоційної тональності фрагментів тексту, що за рахунок модульної інтеграції процедур токенізації на основі підслівних одиниць та визначення формалізованої множини конструктивних параметрів нейромережової моделі, забезпечує адаптацію нейромережових засобів до лінгвістичної специфіки та ресурсних обмежень умов застосування.

- *Вперше* розроблено концептуальну модель розпізнавання емоційної тональності фрагментів тексту, що за рахунок системного узгодження параметрів нейромережевої архітектури з обмеженнями обчислювальних ресурсів та лінгвістичною специфікою навчальних даних, забезпечила формалізований опис напрямків досліджень з розробки відповідних засобів розпізнавання.

- *Отримав подальший розвиток* метод розпізнавання емоційної тональності фрагментів тексту, що за рахунок комплексного використання моделі побудови нейромережових засобів, удосконаленої системи критеріїв оцінювання та розробленого анотованого корпусу україномовних текстів, забезпечує можливість адаптації архітектури засобу розпізнавання до умов застосування та розширення лексичного покриття предметної області, що дозволяє підвищити якість розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

- *Отримала подальший розвиток* система критеріїв оцінювання ефективності засобів розпізнавання емоційної тональності фрагментів тексту, що за рахунок використання розширеної множини метрик якості, інтерпретованості та ресурсоемності, дозволила розрахувати критерії якості класифікації, обчислювальної складності, а на їх основі і критерій сумарної ефективності, що забезпечує комплексного оцінювання засобів розпізнавання з позицій точності, інтерпретованості та обчислювальних витрат.

Практичне значення одержаних результатів полягає у розробці архітектури системи розпізнавання емоційної тональності фрагментів тексту, що дозволяє підвищити якість розпізнавання, адаптована для інтеграції в універсальні комп'ютерні системи моніторингу інформаційного простору, а також може бути використана для створення спеціалізованих інструментальних засобів відповідного призначення.

Практичне цінність одержаних результатів полягає у наступному:

- Застосування розробленого методу розпізнавання емоційної тональності фрагментів тексту дозволяє приблизно на 26% підвищити якість розпізнавання емоційної тональності при розгортанні засобів розпізнавання на високопродуктивних системах або забезпечити функціонування з латентністю менше 100 мкс при розгортанні засобів розпізнавання на мобільних пристроях.

- Створений анотований україномовний корпус даних може бути використаний науковою спільнотою для подальших досліджень у галузі обробки природної мови.

- Розроблені програмні застосунки, що реалізують запропонований метод, впроваджені в навчальний процес на кафедрі системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" (акт впровадження від 30.04.2026).

У вступі показано актуальність теми дисертації, обґрунтовано мету і завдання дослідження, визначено наукову та практичну цінність отриманих результатів і особистий внесок автора у спільних публікаціях.

У першому розділі проведено комплексний аналіз предметної області, розглянуто постановку задачі розпізнавання емоційної тональності в контексті універсальних комп'ютерних систем, проаналізовано відомі засоби та виявлено ключові проблеми, зокрема, дефіцит даних для української мови та відсутність системних підходів до проєктування обчислювально ефективних моделей. Визначено, що важливим напрямком підвищення ефективності є вдосконалення нейромережових засобів розпізнавання.

У другому розділі вдосконалено методологічну базу розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах. Розроблено концептуальну модель розпізнавання емоційної тональності фрагментів тексту. Удосконалено систему критеріїв оцінювання ефективності засобів розпізнавання. Розроблено модель побудови нейромережових засобів розпізнавання емоційної тональності.

У третьому розділі запропоновано метод розпізнавання емоційної тональності текстових фрагментів з використанням нейромережевих засобів в універсальних комп'ютерних системах. В рамках розробки методу обґрунтовано процедуру підготовки вхідних даних. Розроблено математичну модель, яка формалізує процес розпізнавання як композицію функцій представлення, вилучення контекстуальних ознак та класифікації. Розроблено механізм вибору архітектури нейромережевої моделі, що забезпечує адаптацію засобів до умов застосування.

У четвертому розділі розроблено та досліджено систему розпізнавання емоційної тональності текстових фрагментів в універсальних комп'ютерних системах. Розроблено архітектуру програмної системи розпізнавання емоційної тональності фрагментів тексту. Створено анотований україномовний корпус емоційно забарвлених текстів. Експериментально підтверджено, що запропоновані рішення дозволяють підвищити ефективність процесу розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах за рахунок підвищення якості розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

Основні результати дисертаційної роботи опубліковано у 10 публікаціях, з яких 2 статті в періодичних виданнях, що проіндексовані у базі даних Scopus, 3 публікації опубліковані у фахових виданнях, включених до переліку наукових фахових видань України з присвоєнням категорії «Б», 3 публікації у матеріалах наукових конференцій. Отримано 2 авторських свідоцтва на комп'ютерні програми.

Ключові слова: модель, метод, нейронні мережі, емоція, алгоритм, машинне навчання, штучний інтелект, обробка природньої мови, розпізнавання, класифікація, тональне забарвлення, велика мовна модель, глибоке навчання, комп'ютерне моделювання, інтелектуальна система.

ABSTRACT

Korovii O.S. Models and Method for Emotion Recognition of Text Fragments in Universal Computer Systems. – Qualifying scientific work on the rights of a manuscript.

Dissertation for the degree of Doctor of Philosophy in the field of knowledge 12 Information Technologies, specialty 123 Computer Engineering. – National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, 2026.

The dissertation is devoted to solving a scientific problem that consists in developing neural network models and a method aimed at improving the efficiency of recognizing the emotional tone of text fragments in universal computer systems under conditions of limited computational resources, while taking into account the linguistic specificity of the Ukrainian language.

The relevance of this scientific problem is determined by the unprecedented growth in the volume of unstructured textual data in social networks, media, and online platforms, which creates a high demand for automatic emotional tone recognition technologies from government institutions and business organizations. For the Ukrainian information space, this issue acquires strategic importance in the context of the full-scale armed invasion, when the adversary’s information and psychological operations deliberately use emotional manipulation through social networks to destabilize public sentiment and undermine the resilience of the state.

At the same time, most existing tools for recognizing emotional tone require substantial adaptation to the linguistic specificity of the Ukrainian language, are characterized by high computational resource requirements, and depend on foreign infrastructure, which limits their reliability and the credibility of their results. This determines the feasibility of developing neural network models and a method focused on efficient deployment in domestic information systems under conditions of limited computational resources.

The object of research is the processes of emotion recognition of text fragments in universal computer systems.

The subject of research is the models and method for emotion recognition of text fragments.

The aim of the dissertation research is to enhance the efficiency of the emotion recognition process for text fragments in universal computer systems by improving recognition quality while considering computational resource constraints and the linguistic features of the Ukrainian language.

In accordance with the aim of the research, the following scientific objectives have been formulated:

1. To analyze modern methods and tools for recognizing the emotional tone of text and to identify promising ways to improve their efficiency.
2. To further develop the methodological basis for recognizing the emotional tone of text fragments using neural network tools in universal computer systems.
3. To develop a method for recognizing the emotional tone of text fragments using neural network tools in universal computer systems.
4. To develop and investigate a system for recognizing the emotional tone of text fragments in universal computer systems.

The main scientific results of the dissertation research:

- *For the first time, a model for constructing neural network tools* for emotion recognition of text fragments has been developed. Through the modular integration of tokenization procedures based on sub word units and the definition of a formalized set of constructive parameters for the neural network model, it ensures the adaptation of neural network tools to the linguistic specifics and resource constraints of the application environment.
- *For the first time, a conceptual model* for emotion recognition of text fragments has been developed. By systematically aligning neural network architecture parameters with computational resource constraints and the linguistic specifics of training data, it provided a formalized description of research directions for developing appropriate recognition tools.
- *The method for emotion recognition* of text fragments has been *further developed*. Through the comprehensive use of the construction model for neural

network tools, an improved system of evaluation criteria, and a developed annotated corpus of Ukrainian texts, it ensures the possibility of adapting the recognition tool architecture to application conditions and expanding the lexical coverage of the subject domain. This allows for improved recognition quality while considering computational resource constraints and the linguistic features of the Ukrainian language.

- *The system of criteria for evaluating the efficiency* of emotion recognition tools for text fragments has been *further developed*. By utilizing an extended set of metrics for quality, interpretability, and resource intensity, it allowed for the calculation of criteria for classification quality and computational complexity, and based on these, a criterion of total efficiency. This ensures the possibility of a comprehensive evaluation of recognition tools from the standpoints of accuracy, interpretability, and computational costs.

The practical significance of the obtained results lies in the development of an architecture for a system designed to recognize the emotional tone of text fragments. This architecture makes it possible to improve recognition quality, is adapted for integration into universal computer systems for monitoring the information space, and can also be used to create specialized software tools for the corresponding purpose.

The practical value of the obtained results is as follows:

- Application of the developed method for emotion recognition of text fragments allows for an *increase in recognition quality by approximately 26%* when deploying recognition tools on high-performance systems, or ensures functioning with latency of less than 100 μ s when deploying recognition tools on mobile devices.

- The created annotated Ukrainian-language data corpus can be used by the scientific community for further research in the field of Natural Language Processing (NLP).

- The developed software applications implementing the proposed method have been introduced into the educational process at the Department of System

Programming and Specialized Computer Systems of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute” (act of implementation dated 2026).

The *introduction* demonstrates the relevance of the dissertation topic, substantiates the aim and tasks of the research, defines the scientific and practical value of the obtained results, and outlines the author's personal contribution to joint publications.

The *first chapter* provides a comprehensive analysis of the subject area, considers the problem statement of emotion recognition in the context of universal computer systems, analyzes known tools, and identifies key problems, particularly the deficit of data for the Ukrainian language and the lack of systematic approaches to designing computationally efficient models. It is determined that a crucial direction for increasing efficiency is the improvement of neural network recognition tools.

The *second chapter* improves the methodological basis for emotion recognition of text fragments using neural network tools in universal computer systems. A conceptual model for emotion recognition of text fragments is developed. The system of criteria for evaluating the efficiency of recognition tools is improved. A model for constructing neural network tools for emotion recognition is developed.

The *third chapter* proposes a method for emotion recognition of text fragments using neural network tools in universal computer systems. Within the framework of the method development, the procedure for input data preparation is substantiated. A mathematical model is developed that formalizes the recognition process as a composition of functions for representation, contextual feature extraction, and classification. A mechanism for selecting the neural network model architecture is developed, ensuring the adaptation of tools to application conditions.

The *fourth chapter* describes the development and investigation of the system for emotion recognition of text fragments in universal computer systems. The architecture of the software system for emotion recognition of text fragments is developed. An annotated Ukrainian-language corpus of emotionally charged texts is

created. It is experimentally confirmed that the proposed solutions allow for increasing the efficiency of the emotion recognition process in universal computer systems by improving recognition quality while considering computational resource constraints and the linguistic features of the Ukrainian language.

The main results of the dissertation are published in 10 publications, of which 2 are articles in periodicals indexed in the Scopus database, 3 are publications in professional journals included in the list of scientific professional publications of Ukraine (Category "B"), 3 are publications in the proceedings of international scientific conferences, 2 copyright certificates for computer programs have been obtained.

Keywords: model, method, neural networks, emotion, algorithm, machine learning, artificial intelligence, natural language processing, recognition, classification, sentiment analysis, large language model, deep learning, computer modeling, intellectual system.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Статті у фахових виданнях України категорії «Б»:

1. Дичка І.А., Терейковський І.А., Коровій О.С., Терейковська Л.О., Романкевич В.О. Оцінка ефективності засобів розпізнавання емоційної тональності фрагментів тексту. Вчені записки ТНУ імені В.І. Вернадського. Серія: технічні науки. Том 34 № 3, 2023. DOI: <https://doi.org/10.32782/2663-5941/2023.3.1/20>. *Здобувач сформулював постановку задачі оцінювання ефективності засобів розпізнавання емоційної тональності фрагментів тексту, виконав аналіз відомих рішень, сформував базовий перелік характеристик і критеріїв ефективності, здійснив порівняльне оцінювання засобів розпізнавання за бінарною шкалою та узагальнив результати дослідження; І. Дичка забезпечив загальне методологічне керівництво дослідженням; І. Терейковський перевінив наукову обґрунтованість і логічну узгодженість запропонованого підходу до оцінювання; Л. Терейковська оцінила практичну придатність визначених критеріїв для задач аналізу емоційної тональності тексту; В. Романкевич взяв участь в аналізі сучасних засобів розпізнавання емоційної тональності та інтерпретації результатів їх порівняльного оцінювання.*

2. Коровій, О., Терейковський, І. Концептуальна модель процесу визначення емоційної тональності тексту. Комп'ютерно-інтегровані технології: освіта, наука, виробництво, (55), 115-123 DOI: <https://doi.org/10.36910/6775-2524-0560-2024-55-14>. *Здобувач розробив концептуальну модель процесу визначення емоційної тональності тексту, визначив її структуру та основні етапи реалізації, узагальнив чинники, що впливають на ефективність розпізнавання, і підготував текст статті; І.А. Терейковський забезпечив наукове керівництво дослідженням, перевінив логічну узгодженість моделі та її відповідність поставленій науковій задачі.*

3. Коровій, О. С., Терейковський, І. А. Метод побудови нейромережових засобів розпізнавання емоційного забарвлення текстів в універсальних комп'ютерних засобах. Комп'ютерно-інтегровані технології:

освіта, наука, виробництво, (60), 2025. 182-189. <https://doi.org/10.36910/6775-2524-0560-2025-60-19>. *Здобувач розробив метод побудови нейромережових засобів розпізнавання емоційного забарвлення текстів, визначив етапи кодування та декодування даних, формалізував вибір архітектури залежно від ресурсних обмежень і вимог до точності, а також узагальнив підхід до побудови легковагових, рекурентних і трансформерних моделей; І.А. Терейковський забезпечив постановку задачі, наукове керівництво дослідженням, перевірів логічну узгодженість запропонованого методу та його відповідність поставленій науковій задачі.*

Статті у періодичних наукових виданнях, проіндексованих у базах даних Web of Science Core Collection та/або Scopus:

4. Korovii, O., Petrashenko, A. (2022). Adaptation of Distilling Knowledge Method in Natural Language Processing for Sentiment Analysis. In: Hu, Z., Petoukhov, S., Yanovsky, F., He, M. (eds) Advances in Computer Science for Engineering and Manufacturing. ISEM 2021. Lecture Notes in Networks and Systems, vol 463. Springer, Cham. DOI: https://doi.org/10.1007/978-3-031-03877-8_15. *Здобувач адаптував метод дистиляції знань до задачі аналізу емоційної тональності текстів українською та російською мовами, обґрунтував підхід до зменшення обчислювальних витрат без істотної втрати точності, визначив особливості перенесення обчислень із GPU на CPU та узагальнив результати застосування методу для задач sentiment analysis; Петрашенко А. брав участь у постановці дослідження, перевірці коректності адаптації методу та аналізі отриманих результатів.*

5. Tereikovskiy I. A., Korovii O. S. "A Model for Constructing Neural Network Systems Forrecognizing Emotions of Text Fragments" Publ. Nauka i Tekhnika. Odesa: Ukraine. Herald of Advanced Information Technology 8 (2), 197-208. DOI: <https://doi.org/10.15276/hait.08.2025.12>. *Здобувач розробив модель побудови нейромережових засобів розпізнавання емоцій фрагментів тексту, визначив її основні структурні компоненти, узагальнив етапи формування,*

навчання та застосування таких засобів, а також обґрунтував особливості їх використання для обробки текстових даних; Терейковський І.А. забезпечив наукове керівництво дослідженням, перевінив логічну узгодженість запропонованої моделі та її відповідність поставленій науковій задачі.

Публікація у матеріалах наукової конференції:

6. В. Г. Зайцев, О. С. Коровій. Проблематика розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах. IX Міжнародна науково-технічна Internet-конференція “Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами” К: НУХТ, 2022. *Здобувач презентував: аналіз проблематики розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах, основні труднощі реалізації відповідних засобів, ключові чинники, що впливають на точність і практичну придатність таких рішень; В. Г. Зайцев брав участь у постановці дослідження, обговоренні проблемних аспектів застосування методів аналізу тексту та редагуванні матеріалів публікації.*

7. Терейковський І.А., Коровій О.С. Обґрунтування актуальності досліджень в області розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах. Матеріали XV науково-практичної конференції магістрантів та аспірантів «Прикладна математика та комп'ютинг», 16-18 листопада 2022. – К: Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2022 С. 361-364. *Здобувач презентував актуальність досліджень у сфері розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах та підготував текст публікації; Терейковський І.А. забезпечив наукове керівництво дослідженням, брав участь у постановці задачі, обґрунтуванні наукової актуальності роботи та редагуванні матеріалів публікації.*

8. I. Tereikovskiy, O. Korovii. Conceptual model of the process for determining the emotional coloring and tonality of text fragments. Матеріали XVII науково-практичної конференції магістрантів та аспірантів «Прикладна математика та комп'ютинг», 20-22 листопада 2024. – К: Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2024. С.478-482. *Здобувач презентував концептуальну модель процесу визначення емоційного забарвлення та тональності фрагментів тексту; Терейковський І.А. забезпечив наукове керівництво дослідженням, перевірів логічну узгодженість запропонованої моделі та її відповідність поставленій науковій задачі.*

Авторські свідоцтва на комп'ютерні програми:

9. Свідоцтво про авторське право на службовий твір (комп'ютерна програма «Text analysis») - 7087 від 15.04.2024. Автори: Терейковський І.А., Коровій О.С., Дичка І.А., Романкевич В.О., Тарасенко-Клятченко О.В. *Здобувач розробив основні програмні модулі комп'ютерної програми «Text analysis», що забезпечують аналіз текстових фрагментів і визначення їх емоційної тональності, виконав налаштування функціональних компонентів і тестування працездатності програми; Терейковський І.А. надав методичні рекомендації щодо побудови архітектури програмного забезпечення та організації процесу розроблення; Дичка І.А. структурував методичні засади реалізації програми та результати її випробування; Романкевич В.О. структурував технічні характеристики й алгоритми роботи програми; Тарасенко-Клятченко О.В. здійснювала технічне та методичне редагування документації.*

10. Свідоцтво про авторське право на службовий твір (комп'ютерна програма «Emotion Recognition») – 7433 від 14.01.2025. CR3610050325. Автори: Терейковський І.А., Коровій О.С., Дичка І.А., Романкевич В.О., Тарасенко-Клятченко О.В. *Здобувач розробив основні програмні модулі комп'ютерної програми «Emotion Recognition», що забезпечують*

автоматизоване розпізнавання емоційного забарвлення текстових фрагментів, виконав налаштування функціональних компонентів і тестування працездатності програми; І.А. Терейковський надав методичні рекомендації щодо побудови архітектури програмного забезпечення та організації процесу розроблення; Дичка І.А. структурував методичні засади реалізації програми та результати її випробування; Романкевич В.О. структурував технічні характеристики й алгоритми роботи програми; Тарасенко-Клятченко О.В. здійснювала технічне та методичне редагування документації.

ЗМІСТ

АНОТАЦІЯ	2
ABSTRACT	7
СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА	11
ЗМІСТ	17
СПИСОК УМОВНИХ СКОРОЧЕНЬ	19
ВСТУП	20
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОЦІНКА ЕФЕКТИВНОСТІ ЗАСОБІВ РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ	26
1.1 Постановка задачі розпізнавання емоційної тональності в універсальних комп'ютерних системах	26
1.2 Аналіз відомих засобів розпізнавання емоційної тональності	29
1.3 Оцінка перспектив підвищення ефективності розпізнавання емоційного забарвлення фрагментів текстів українською мовою	47
1.4 Висновки до першого розділу	50
2 ЕЛЕМЕНТИ МЕТОДОЛОГІЧНОГО ЗАБЕЗПЕЧЕННЯ РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ	52
2.1 Концептуальна модель розпізнавання емоційної тональності текстових фрагментів	52
2.2 Система критеріїв оцінювання ефективності засобів розпізнавання	66
2.3 Модель побудови нейромережових засобів розпізнавання емоційного забарвлення фрагментів текстів	71
2.4 Висновки до другого розділу	88
3 МЕТОД РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ	90

		18
3.1	Процедура формування вхідних і вихідних даних	90
3.2	Аналітичне забезпечення методу	101
3.3	Етапи методу розпізнавання емоційної тональності	107
3.4	Висновки до третього розділу	121
4	СИСТЕМА РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ	123
4.1	Архітектура системи розпізнавання	123
4.2	Експериментальна установка	127
4.3	Експериментальні дослідження	132
4.4	Висновки до четвертого розділу	146
	ВИСНОВКИ	147
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	149
	ДОДАТОК А. Програмний код застосунку	163
	ДОДАТОК Б. Акти впровадженнь результатів дисертаційного дослідження	171

СПИСОК УМОВНИХ СКОРОЧЕНЬ

ГНМ – глибока нейронна мережа

ІС – інформаційна система

КС – комп’ютерна система

НМЗ – нейромережевий засіб

НММ – нейромережева модель

РЕТ – розпізнавання емоційної тональності

CNN - Convolutional Neural Network

CPU - Central Processing Unit

GPU - Graphics Processing Unit

LLM - Large Language Model

NLP - Natural Language Processing

RNN - Recurrent Neural Network

ВСТУП

Актуальність теми. Сучасний етап розвитку інформаційного суспільства характеризується безпрецедентним зростанням обсягів неструктурованих текстових даних, що генеруються в соціальних мережах, медіа та на онлайн-платформах. Цей колосальний масив інформації є не лише сховищем фактів, але й дзеркалом суспільних настроїв, індивідуальних переживань та колективних емоцій. У цьому контексті розпізнавання емоційної тональності (PET) фрагментів тексту виступає як ключова технологія обробки природної мови, що дозволяє автоматично вилучати, ідентифікувати та класифікувати суб'єктивну інформацію з текстових фрагментів. Глобальний ринок PET демонструє значне зростання і, за прогнозами, досягне 6.4 мільярда доларів США до 2027 року, що свідчить про високий попит на такі технології з боку державних органів та бізнес-структур [1-3].

Для українського мовного простору актуальність проблематики істотно зростає в умовах повномасштабного збройного вторгнення та посилення інформаційно-психологічних впливів на український медіапростір. Інформаційно-психологічні спеціальні операції противника активно застосовують соціальні мережі для поширення дезінформації та маніпулятивних наративів, спрямованих на дестабілізацію суспільних настроїв і підірив стійкості держави. Одним із ключових механізмів такого впливу є емоційна маніпуляція, що передбачає цілеспрямоване формування негативних емоційних реакцій аудиторії. У цьому контексті створення ефективних інструментів для автоматизованого PET текстового контенту набуває стратегічного значення. Вирішення цієї задачі узгоджується з пріоритетами, визначеними Стратегією національної безпеки України (2020), Стратегією кібербезпеки України (2021) та Стратегією інформаційної безпеки України (2021), які визначають необхідність підвищення стійкості держави до інформаційно-психологічних впливів і зміцнення інформаційного

суверенітету. Зазначимо, що значущість підвищення ефективності інструментів автоматизованого моніторингу інформаційного простору підтверджується Розпорядженням Кабінету Міністрів України від 7 березня 2025 р. № 204-р, яким затверджено план заходів на 2025 рік з реалізації Стратегії кібербезпеки України та підкреслюється важливість удосконалення технологічних механізмів виявлення інформаційних загроз та підвищення ефективності реагування на них. Разом із тим, впровадження відомих промислових засобів РЕТ текстових фрагментів у вітчизняні інформаційні системи (ІС) потребує їх складної адаптації до варіативності умов застосування, пов'язаних із лексичним покриттям моделей, доповненням навчальних баз даних специфічним тематичним контентом, величиною допустимої похибки розпізнавання, швидкодією обробки великих потоків текстових даних та ресурсними обмеженнями при створенні і функціонуванні систем. Також недоліками поширених засобів РЕТ є висока вартість, відсутність детальної науково-технічної документації та недостатня надійність, пов'язана з використанням закордонних інформаційних ресурсів, що може викликати сумніви у достовірності отриманих результатів. Разом з тим аналіз наукових праць провідних вітчизняних та закордонних вчених Н. Д. Панкратової, Д. В. Ланде, С. Г. Стіренка, І. А. Терейковського, Є. С. Сулеми, В. В. Пасічника, В. А. Широкова, Н. П. Дарчук, Б. Лю (B. Liu), Е. Камбрія (E. Cambria), Р. Пікард (R. Picard) та С. Порія (S. Poria) у сфері системного аналізу інформаційних процесів та РЕТ текстових фрагментів свідчить про високий рівень досягнень у фундаментальних аспектах комп'ютерної лінгвістики та підтверджує високу перспективність використання нейромережевих технологій. Разом з тим, попри значний прогрес у підвищенні точності розпізнавання, недостатня увага приділена питанням розробки моделей і методів, орієнтованих на адаптацію побудованих на їх основі засобів РЕТ до розгортання в універсальних комп'ютерних системах, де критично важливими є швидкодія та обчислювальні ресурси. Цим пояснюється актуальність **наукового завдання дослідження**, яке полягає у розробці нейромережевих

моделей та методу, що забезпечують підвищення ефективності розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах в умовах обмежених обчислювальних ресурсів з урахуванням лінгвістичної специфіки української мови.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційна робота виконана за планами наукових і науково-технічних робіт Національного технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" та в рамках науково-дослідної роботи «Методи, моделі та засоби моніторингу закликів до тероризму у онлайн соціальних мережах» (№ 0124U003866).

Мета і завдання дослідження. Метою дисертаційного дослідження є підвищення ефективності засобів розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах за рахунок підвищення якості розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

Для досягнення поставленої мети в роботі було вирішено такі завдання:

1. Провести аналіз сучасних методів і засобів розпізнавання емоційної тональності тексту та визначити перспективні шляхи підвищення їх ефективності.

2. Провести розвиток методологічної бази розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах.

3. Розробити метод розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах.

4. Розробити та дослідити систему розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах.

Об'єкт дослідження – процеси розпізнавання емоційної тональності текстових фрагментів в універсальних комп'ютерних системах.

Предмет дослідження – моделі та метод розпізнавання емоційної тональності текстових фрагментів.

Методи дослідження. Використано методи системного аналізу (для побудови концептуальної моделі системи); теорії штучних нейронних мереж (для розробки нейромережових моделей розпізнавання); математичного моделювання (для формалізації методу розпізнавання); комп'ютерної лінгвістики (для попередньої обробки та токенизації україномовних текстів); планування експерименту та статистичного аналізу (для організації експериментальних досліджень, обробки результатів та оцінювання ефективності розроблених засобів).

Наукова новизна одержаних результатів полягає у наступному:

- Вперше розроблено модель реалізації нейромережових засобів розпізнавання емоційної тональності фрагментів тексту, що за рахунок модульної інтеграції процедур токенизації на основі підслівних одиниць та визначення формалізованої множини конструктивних параметрів нейромережової моделі, забезпечує адаптацію нейромережових засобів до лінгвістичної специфіки та ресурсних обмежень умов застосування.

- Вперше розроблено концептуальну модель розпізнавання емоційної тональності фрагментів тексту, що за рахунок системного узгодження параметрів нейромережової архітектури з обмеженнями обчислювальних ресурсів та лінгвістичною специфікою навчальних даних, забезпечила формалізований опис напрямків досліджень з розробки відповідних засобів розпізнавання.

- Отримав подальший розвиток метод розпізнавання емоційної тональності фрагментів тексту, що за рахунок комплексного використання моделі побудови нейромережових засобів, удосконаленої системи критеріїв оцінювання та розробленого анотованого корпусу україномовних текстів, забезпечує можливість адаптації архітектури засобу розпізнавання до умов застосування та розширення лексичного покриття предметної області, що дозволяє підвищити якість розпізнавання з урахуванням обмежень щодо

обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

- Отримала подальший розвиток система критеріїв оцінювання ефективності засобів розпізнавання емоційної тональності фрагментів тексту, що за рахунок використання розширеної множини метрик якості, інтерпретованості та ресурсоемності, дозволила розрахувати критерії якості класифікації, обчислювальної складності, а на їх основі і критерій сумарної ефективності, що забезпечує можливість комплексного оцінювання засобів розпізнавання з позицій точності, інтерпретованості та обчислювальних витрат.

Практичне значення одержаних результатів. Запропоновані моделі та метод дозволили розробити архітектуру системи розпізнавання емоційної тональності фрагментів тексту, що дозволяє підвищити якість розпізнавання, адаптована для інтеграції в універсальні комп'ютерні системи моніторингу інформаційного простору, а також може бути використана для створення спеціалізованих інструментальних засобів відповідного призначення.

Практична цінність полягає у наступному:

- Застосування розробленого методу розпізнавання емоційної тональності фрагментів тексту дозволяє на 26% підвищити якість розпізнавання емоційної тональності для умов коли засоби розпізнавання розгорнуто на високопродуктивних системах або забезпечити функціонування з латентністю менше 100 мкс при розгортанні засобів розпізнавання на мобільних пристроях.

- Створений анотований україномовний корпус даних може бути використаний науковою спільнотою для подальших досліджень у галузі обробки природної мови.

- Розроблені програмні застосунки, що реалізують запропонований метод, впроваджені в навчальний процес на кафедрі системного програмування і спеціалізованих комп'ютерних систем Національного

технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" (акт впровадження від 30.04.2026).

Особистий внесок здобувача. Усі основні результати дисертаційного дослідження, які представлені до захисту, одержані автором особисто. У працях за темою дисертації, опублікованих у співавторстві, здобувачі особисто належить: [4] визначено перелік критеріїв ефективності, проведено оцінювання відомих засобів розпізнавання емоційної тональності фрагментів тексту, визначено перспективні шляхи підвищення ефективності засобів розпізнавання; у [5, 6] реалізована формалізація процесу розпізнавання емоційної тональності фрагментів тексту; у [7] розроблено модель розпізнавання емоційної тональності фрагментів тексту; у [8, 9] розроблено метод побудови нейромережових засобів розпізнавання емоційного забарвлення текстів в універсальних комп'ютерних засобах; у [6, 10] обґрунтовано актуальність наукового завдання розпізнавання емоційного забарвлення текстів в універсальних комп'ютерних засобах; у [11, 12] розробка комп'ютерних програм.

Апробація результатів дисертації. Основні результати дисертаційного дослідження доповідалися на 3 міжнародних та всеукраїнських науково-практичних конференціях, зокрема: IX Міжнародна науково-технічна Internet-конференція “Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами” (Київ, 2022); Прикладна математика та комп'ютинг ПМК' 2022 (Київ, 2022); Applied mathematics and computing AMC' 2024. (Київ, 2024).

Публікації. Основні результати дисертаційної роботи опубліковано у 5 наукових працях, з яких 3 у фахових вітчизняних виданнях категорії Б та 2 у періодичних виданнях, проіндексованих у базі даних Scopus.

Структура і обсяг роботи. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел з 103 найменувань, 2 додатки на 10 сторінках, 27 рисунків, 6 таблиць – всього на 172 сторінках. Основний текст дисертації викладено на 130 сторінках.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОЦІНКА ЕФЕКТИВНОСТІ ЗАСОБІВ РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ

1.1. Постановка задачі розпізнавання емоційної тональності в універсальних комп'ютерних системах

Сучасний етап розвитку інформаційного суспільства характеризується безпрецедентним зростанням обсягів неструктурованих текстових даних, які генеруються користувачами в глобальній мережі [1-3]. Соціальні медіа, платформи для відгуків, новинні портали, блоги та форуми щомиті наповнюються мільйонами повідомлень, що формують так званий "цифровий слід" людства [8, 13, 14]. Цей масив інформації є не лише сховищем фактів, але й суспільних настроїв, індивідуальних переживань та колективних емоцій. У цьому контексті розпізнавання емоційної тональності тексту виступає як ключова технологія обробки природної мови (англ. Natural Language Processing, NLP), що дозволяє автоматично вилучати, ідентифікувати та класифікувати суб'єктивну інформацію з текстових фрагментів. Класична постановка задачі передбачає класифікацію тексту за полярністю (позитивний, негативний, нейтральний), однак сучасні підходи розширюють цей спектр до більш гранульованого аналізу, що включає ідентифікацію конкретних емоцій (радість, гнів, сум, страх), аналіз аспектно-орієнтованої тональності (англ. Aspect-Based Sentiment Analysis, ABSA), а також виявлення сарказму та іронії [15, 16]. Актуальність цієї задачі зумовлена її надзвичайно широким прикладним потенціалом, що охоплює моніторинг репутації брендів, аналіз клієнтського досвіду, прогнозування фінансових ринків, оцінку суспільно-політичних процесів та розробку адаптивних людино-машинних інтерфейсів.

Для українського мовного простору задача автоматизованого розпізнавання емоційної тональності набуває особливої ваги та стикається з низкою унікальних викликів. Українська мова, як і багато інших, належить до категорії "низькоресурсних" (англ. low-resource languages) в галузі NLP, що означає суттєвий дефіцит високоякісних лінгвістичних ресурсів порівняно з англійською [17]. Критично бракує великих, збалансованих та публічно доступних анотованих корпусів текстів, які є фундаментальною передумовою для навчання ефективних моделей на основі керованого машинного навчання [8, 14]. Створення таких корпусів є трудомістким та вартісним процесом, що вимагає залучення кваліфікованих лінгвістів-анотаторів. Додаткові труднощі створює висока морфологічна складність української мови (велика кількість відмінкових та дієслівних форм), що призводить до проблеми розрідженості даних (англ. data sparsity) і ускладнює генералізацію моделей. Більше того, неформальне онлайн-спілкування в українському сегменті інтернету часто характеризується такими явищами, як використання суржику, перемикування кодів (англ. code-switching) з російською чи англійською мовами та вживання ненормативної лексики, що створює значні перешкоди для традиційних NLP-моделей. Таким чином, розробка моделей, стійких до цих лінгвістичних особливостей, є нагальною науково-технічною проблемою, вирішення якої є критично важливим для забезпечення повноцінної присутності української мови в сучасному цифровому світі.

Надзвичайної гостроти ця проблематика набуває в контексті повномасштабної гібридної війни. Ворожі інформаційно-психологічні спеціальні операції (ІПСО) активно використовують соціальні мережі та медіа для поширення дезінформації, маніпулятивних наративів та пропаганди з метою поляризації суспільства, деморалізації населення та підриву довіри до державних інституцій. Ключовим інструментом такої діяльності є емоційна маніпуляція - цілеспрямоване розпалювання страху, ненависті, зневіри та панічних настроїв. Автоматизований моніторинг та аналіз емоційної тональності текстового контенту в режимі реального часу дозволяє оперативно

виявляти скоординовані інформаційні атаки, ідентифікувати джерела ворожої пропаганди та відстежувати динаміку суспільних настроїв у відповідь на ті чи інші події. Розробка таких інструментів, адаптованих до специфіки українського медіа-ландшафту, є не просто академічною задачею, а стратегічним пріоритетом для захисту інформаційного суверенітету та зміцнення національної безпеки України.

Водночас, ефективне вирішення зазначених завдань наштовхується на фундаментальне обмеження, пов'язане з апаратними ресурсами. Сучасні моделі обробки природної мови, що демонструють найвищу точність, зокрема архітектури на основі механізму самоуваги - трансформери (наприклад, BERT[18], RoBERTa[19], GPT[20]), є надзвичайно обчислювально затратними. Вони містять сотні мільйонів або навіть мільярди параметрів, вимагають для свого навчання та функціонування потужних графічних процесорів або тензорних процесорів і споживають значні обсяги енергії та води для охолодження [4, 6]. Це створює "обчислювальний бар'єр", що унеможливорює їхнє широке застосування в універсальних комп'ютерних системах, які охоплюють не лише високопродуктивні хмарні сервери, але й звичайні настільні ПК, мобільні пристрої, вбудовані системи та пристрої Інтернету речей (англ. Internet of Things, IoT). Багато потенційних користувачів таких систем, зокрема державні установи, громадські організації та невеликі компанії, не мають доступу до необхідної дороговартісної інфраструктури. Отже, виникає гостра науково-прикладна проблема розробки таких моделей та методів, які б забезпечували оптимальний компроміс між точністю прогнозування, швидкістю роботи та вимогами до обчислювальних ресурсів. Проте, існуючі підходи до вирішення проблеми обчислювальної ефективності, хоч і є дієвими, часто мають фрагментарний характер. Такі методи, як дистиляція знань (англ. knowledge distillation), квантизація (англ. quantization) та прунінг (англ. pruning), зазвичай застосовуються як техніки постфактум оптимізації до вже існуючих, переважно громіздких архітектур [21-23]. Такий підхід не гарантує досягнення оптимального балансу між точністю та

продуктивністю, оскільки ефективність стає вторинною ціллю, а не фундаментальним принципом, закладеним у процес розробки моделі. Це призводить до ситуації, коли дослідники змушені застосовувати комбінації евристичних рішень, що ускладнює відтворюваність результатів та не дозволяє створити єдину, системну методологію для побудови ефективних рішень для різнорідних апаратних платформ.

У цьому контексті перспективним напрямом дослідження вбачається фундаментальний методологічний зсув: перехід від застосування розрізнених технік оптимізації до розробки та формалізації комплексного методу. Такий метод має описувати узагальнені та послідовні етапи створення моделей, що здатні ефективно вирішувати двоєдину задачу аналізу емоційного забарвлення тексту. Ця задача поєднує, по-перше, визначення загальної емоційної тональності (полярності) тексту (позитивна, негативна, нейтральна), що дає загальне уявлення про ставлення автора. По-друге, що є не менш важливим, вона включає ідентифікацію в тексті більш гранульованих, дискретних категорій емоцій (наприклад, радість, сум, гнів, страх), що дозволяє отримати значно глибше розуміння психологічного стану та мотивації автора [24, 25]. Такий комплексний підхід є критично важливим для глибокого аналізу інформаційних маніпуляцій, де часто використовуються складні емоційні тригери.

Хоча багато алгоритмів та методів створені для цієї мети, їхня ефективність часто недостатня. Оцінювання цих методів та урахування мовних варіацій може покращити розпізнавання емоційної тональності.

1.2. Аналіз відомих засобів розпізнавання емоційної тональності

Попри значний масив публікацій [24-33], у проаналізованій літературі відсутній узгоджений і методологічно обґрунтований перелік характеристик систем РЕТ, якому ці системи мають відповідати для коректного розв'язання поставленої задачі. Ідеться не лише про вибір архітектури, а про цілісну сукупність вимог:

1. Семантичне покриття (які саме емоційні категорії або континуальні виміри – валентність/активація/домінантність – підтримує система; чи допускається мульти-мітковість та неоднозначність).

2. Контекстну чутливість (здатність інтерпретувати прагматичні й дискурсивні сигнали, іронію, сарказм, метафору, емодзі, пунктуаційні та орфографічні маркери інтенсивності).

3. Лінгвістичну робастність (стійкість до морфологічної варіативності, просторіччя, діалектів, змішування мов і доменної зсувності).

4. Метрики та верифікаційні властивості (калібрування ймовірностей, узгодженість анотацій – коефіцієнт міжанотаторської згоди, придатність до крос-доменного узагальнення, макро/мікро-F1, AUC).

5. Обчислювальну ефективність (затримка, пропускна здатність, пам'ять/енерговитрати, можливість розгортання на «універсальних» обчислювальних платформах від мобільних до GPU-кластерів).

6. Інтерпретованість і відтворюваність (простежуваність рішень, помилок, відкритість даних і коду) та етичні аспекти (упередження, справедливість щодо соціолінгвістичних груп, механізми контролю ризиків).

Належна адаптація перелічених характеристик до задачі визначення емоційного забарвлення тексту є вирішальною для практичної придатності методів, оскільки саме вона забезпечує баланс між точністю, пояснюваністю та вартістю експлуатації. Змістовно це охоплює систематичне вивчення емоційних відтінків, закодованих у лексичних одиницях і фразеологізмах, урахування локального й глобального контексту, а також моделювання дискурсивних взаємодій. У дискретному підході зазвичай виокремлюють шість базових емоцій - радість, смуток, гнів, страх, відразу та здивування-тоді як у задачах сентимент-аналізу переважає тріада позитивної, негативної й нейтральної тональності; на практиці ці множини гнучко розширюють додатковими емоційними станами й інтенсивностями, що вимагає чіткої специфікації схем анотації та прозорого відображення.

Одним із підходів до розпізнавання емоційної тональності, можна виокремити підходи, які розпізнають в тексті поклики до емоцій. У статті [26] наведено близько вісімнадцяти автоматично обчислюваних індикаторів, спрямованих на виявлення дезінформації та маніпулятивних практик у текстах. Автори виходять із припущення, що обман і вплив на читача мають виразні «поверхневі» сигнали, які проявляються у поєднанні лексичних, стилістичних і дискурсивних характеристик контенту. Ідеться, зокрема, про лексико-семантичні ознаки (полярність та інтенсивність сентименту; частка емоційно маркованої лексики; наявність модальних і епістемічних маркерів на кшталт «мабуть», «очевидно»; перевага оцінних прикметників і інтенсифікаторів), синтаксично-дискурсивні патерни (частотність питальних/окличних конструкцій; риторичні запитання; узагальнювальні формули «всі», «ніхто»; дейктика та апеляції до неозначених «джерел»), стиліметричні профілі (лексична різноманітність, середня довжина речення, показники читабельності, відсоток повторів), а також прагматичні/паралінгвістичні маркери (підсилена пунктуація, великі літери для емпізи, багатокрапки, надмірна експресивність). Додатково розглядаються індикатори суб'єктивності й упевненості/невизначеності висловлювань, що у сукупності дозволяє моделювати емоційний відгук і «навантаженість» мови. Такий набір ознак є привабливим завдяки інтерпретованості та технологічній портативності, однак водночас чутливий до доменної зсувності, культурно-мовних відмін і адаптивної поведінки злоумисників (імітація нейтрального стилю, саркастичне кодування тощо).

Відтак, як зазначено в [26], ці контент-орієнтовані індикатори доцільно поєднувати з більш глибинними семантичними перевірками (співставлення тверджень із базами знань, факт-чекінг), метаданими джерел і мережевими сигналами поширення, а для низькоресурсних мов-локалізувати словники, навчити моделі на репрезентативних корпусах, калібрувати пороги прийняття рішень і забезпечувати пояснюваність через ваги ознак та приклади-прототипи.

Стаття [27] фокусується на виявленні причини (стимулу) емоції як на окремій підзадачі емоційного аналізу й пропонує зіставлення двох формалізацій: маркування послідовності токенів (тегування - “які саме слова описують стимул?”) та класифікацію всього речення (“чи містить дане речення стимул емоції?”). Автори показують, що врахування спільних лінгвістичних і стилістичних ознак (лексичної полярності й інтенсивності, модальності, дискурсивних маркерів тощо) у поєднанні з явним виявленням тригера події/ситуації, яка викликає емоцію, покращує інтерпретованість і дає змогу будувати більш точні пайплайни розпізнавання емоційної тональності. Методично запропоновано інтегровану рамку для чесного порівняння підходів і випробувано її на чотирьох англomовних датасетах із різних доменів; результати демонструють перевагу маркування послідовності токенів у трьох із чотирьох наборів як за токен-, так і за реченьо-орієнтованими метриками, тоді як класифікація речень виграє лише на корпусі з високою щільністю анотацій.

Помилковий аналіз додатково свідчить, що для англійської, речення часто не є адекватною одиницею стимулу (через варіативні межі речень, багатослівні вирази, вкладені залежності), тож більш дрібна одиниця - токен/фраза - краще захоплює причинний тригер емоції. Таким чином, у ширшому контексті задач емоційного аналізу, інкорпорація детектора стимулу забезпечує тонше моделювання зв'язку між емоційною реакцією та її причиною і може підвищувати якість класифікації емоцій або сентименту, особливо коли схеми анотації чітко погоджують рівень гранулярності (токен/фраза/речення) з цільовими ознаками моделі.

Під час РЕТ КС неминуче стикаються з багатшаровою природою людської мови: підтекстом, культурними референсами, код-світчингом та жанровими конвенціями. Особливі труднощі спричиняють сарказм і іронія - способи вираження, в яких комунікативний намір часто суперечить буквальному змісту. Як підкреслюється в огляді [28], сарказм «маскується» під позитивну або нейтральну лексичну оболонку, тоді як справжня

оцінка/емоція є протилежною; це породжує системні помилки в моделях, що спираються на поверхневі маркери (словники тональності, полярність слів) без урахування дискурсу, світу знань і контекстної невідповідності. Тому сучасні підходи доповнюють звичайний сентимент-аналіз детекторами сарказму, які моделюють:

- конфлікт між «очікуваною» та «спостережуваною» полярністю,
- риторичні структури (риторичні запитання, гіпербола),
- орфографічно-пунктуаційні сигнали (капслок, повтори знаків «?!», багатокрапка),
- емодзі/хештеги як слабкі маркери (#sarcasm),
- авторський та міжповідомлюваний контекст (попередня історія постів, відповіді в треді).

Для низькоресурсних мов важливими є доменна адаптація, локалізація словників, багатомовний трансфер і слабо-контрольоване навчання на соціальних медіа.

Рівні аналізу емоційного забарвлення доцільно розглядати і як окремі задачі, і як ієрархічну композицію:

- Лексичний рівень (окремі вислови/токени). Ідентифікація емоційно маркованих слів/фраз, інтенсифікаторів, заперечень, модальних і епістемічних маркерів, емодзі та ідіом. Критичною є нормалізація (лематизація, правописні варіанти), обробка сленгу й полісемії, а також моделювання області дії заперечення та підсилювачів. Обмеження: брак контексту призводить до хибних інтерпретацій (зокрема для сарказму й метафори).

- Рівень речення. Урахування синтаксису (залежності слів у реченні), прагматики (імплікатура, ввічливість), локального дискурсу (зв'язки «причина-наслідок»). На цьому рівні краще відслідковується зміна полярності між сусідніми реченнями та «перекид» емоції всередині думки; підхід особливо корисний у складних текстах, де тон «пульсує». Дотичні завдання - виявлення стимулу емоції та тригерів подій - можуть підсилювати точність, бо пояснюють «чому» виникла емоція.

- Рівень документа. Моделювання глобальної когерентності, тематичних зсувів і наративної динаміки (емоційні траєкторії по тексту), виявлення суперечностей (позитивний заголовок - негативне тіло), агрегація оцінок із різних сегментів. Це доречно для відгуків, оглядів, довгих постів, де «середня» емоція не збігається з емоційними піками.

Додатково вирізняють аспектно-цільовий рівень (емоція/сентимент щодо конкретної сутності або атрибуту, що особливо важливо при «вибірковому» сарказмі: позитив у цілому, але негатив щодо окремого аспекту) та рівень розмови/треду (контекст попередніх повідомлень, адресат, соціальні ролі), який у соціальних мережах часто визначальний.

Зазвичай ці підходи комбінують разом для покращення точності визначення емоційної тональності. В роботах [29-31], використовують підхід де комбінують визначення на рівнів окремих висловів, а також рівень документів. Спочатку визначається контекстна залежність між словами та фразами, а потім ця інформація передається, разом із всім текстом, в модель для подальшого розпізнавання. Для визначення, які слова та фрази можуть викликати емоції доцільніше використовувати рівень слів та фраз, а також рівень речень, це допомагає точно ідентифікувати емоційний окрас частини тексту [27]. При комбінації різних рівнів використовується також комбінація різних моделей машинного навчання. Таке використання декількох моделей збільшує точність визначення [27, 31, 32]. З інженерного погляду це перегукується з практикою гібридних/ансамблевих схем у суміжних завданнях де поєднання різнотипних ознак і джерел (лінгвістичні, стилістичні, мережеві) стабільно підсилює робастність; подібні принципи корисні й для інтеграції рівнів у сентимент-аналізі [26].

Нині дедалі частіше залучають великі мовні моделі (англ. Large Language Model, LLM) як універсальні «фондові» моделі зі значним обсягом попередньо вивчених знань і сильними здібностями до zero-shot та few-shot розв'язання задач через приклади в підказці (англ. in-context learning) та інструкційне налаштування. Типову класифікацію емоцій/сентименту формулюють як

інструкцію з чітко визначеним простором міток (напр., {позитивний, нейтральний, негативний} або множини емоцій) і, за потреби, додають кілька демонстрацій. Якість такої постановки суттєво залежить від дизайну підказки (вербалізаторів міток, формулювань інструкцій, порядку прикладів) і часто підвищується за рахунок ансамблювання різних підказок та калібрування міток.

Формально близькою до цієї лінії є «adversarial soft prompt tuning» зі статті [33]: база моделі заморожується, а «м'які» підказки (континуальні вектори, що додаються до вхідної послідовності) навчаються так, щоби бути водночас інформативними для класифікації та доменно інваріантними. Для цього вводиться дискримінатор домену і використовується адверсаріальна оптимізація: представлення, з якими працює класифікатор, заохочують «заплутувати» дискримінатор, зменшуючи зсув між джерельними та цільовими доменами. У результаті метод [33] покращує міждоменне розпізнавання емоційної тональності, зберігаючи низьку кількість навчених параметрів і стабільність на невеликих наборах прикладів.

Для формування переліку основних характеристик, що визначають ефективність розпізнавання емоційного забарвлення тексту, було використано методологічний підхід, наведений у дисертації [34], який полягає у системному співвіднесенні критеріїв ефективності з функціональними та нефункціональними вимогами до комп'ютерних засобів. Цей підхід дозволяє перейти від абстрактних показників якості до конкретних архітектурних та алгоритмічних рішень. Базисом для визначення вказаних вимог стали сучасні науково-прикладні роботи в галузі обробки природної мови (NLP) з використанням методів та моделей штучного інтелекту, що дозволило сформулювати комплексну рамку для оцінки та проєктування таких систем.

У [26, 35, 36] докладно описано типовий конвеєр для класифікації текстів про дезінформацію:

- постановка задач (від розрізнення власне «fake news» і маніпуляцій до формалізації підзадач),

- спектр методів (фактчекінг, розпізнавання маніпулятивного стилю, оцінка надійності джерела, аналіз поширення інформації),
- дані та оцінювання.

Ключовий зсув – від ручного табличного подання проєктування ознак (n-грам, стилістичних і риторичних індикаторів тощо) до глибокого навчання, де класифікатор працює безпосередньо з послідовністю токенів і вловлює довгі залежності (на основі глибоких нейронних мереж RNN/LSTM), а також до трансформерів на кшталт BERT для тоншого рівня (речення/слово) із фіксованими мітками маніпуляцій.

Аналіз поширення працює з «каскадами» ретвітів/ланцюжків: показано, що неправдиві повідомлення розповсюджуються «далі, швидше й ширше», особливо в політичному домені, але метод має слабку передбачуваність на ранніх стадіях (потрібно накопичити достатньо взаємодій). Окремий розділ [5] присвячено наборам даних і оцінюванню. Автори наголошують на малих/гетерогенних корпусах і різних схемах анотації, що ускладнює пряме порівняння методів.

Робота [26] фіксує еволюцію від старих методів, які навчали класифікаторів на різних ручних функціях, витягнутих із тексту, до останніх підходів, які використовують методи глибокого навчання, де класифікатор працює безпосередньо з вихідним текстом, з одночасною інтеграцією комплексних сигналів (зміст, стиль, джерело, дифузія) та вимогою ретельної попередньої обробки тексту (від виділення тверджень до нормалізації), а також підкреслює потребу в більш репрезентативних і стандартизованих наборах даних для відтворюваної оцінки.

Таким чином однією із вимог являється необхідність формування корпусу даних, або використання уже відомих. Слід зазначити, що формування означених корпусів є складною та достатньо ресурсоємною процедурою, що вимагає залучення людей-анотаторів. Крім того можна зробити висновок про залежність ефективності комп'ютерних засобів розпізнавання емоційної

тональності фрагментів текстів від можливості використання відомих корпусів.

В статті [31] автори пропонують нову стратегію маркування частин тексту, яка включає два набори міток пари токенів: основний набір міток і повний набір міток. Автори також пропонують ефективну модель, що побудована на основі глибокої нейронної мережі, яка працює з їх стратегією маркування. Ця модель використовує мережі звернення уваги на графи для ітеративного уточнення представлень токенів і адаптивний класифікатор із кількома мітками для динамічного прогнозування кількох зв'язків між парами токенів. Для своїх експериментів автори використовували п'ять контрольних наборів даних чотирма мовами. В статті повідомляється, що їхня модель значно перевершила попередні найсучасніші моделі. Також використовуючи стратегію маркування та ефективну архітектуру моделі, демонструється багатообіцяючі результати в експериментах на кількох наборах даних.

У роботі [28] систематизовано три великі підходи:

- Лексичні/прагматичні ознаки. Моделі спираються на індикатори на кшталт емодзі/емотиконів, повторної пунктуації, нетипової капіталізації, інтенсифікаторів («дуже», «very») і характерних шаблонів фраз; часто додають евристику «інконгруентності» між позитивною лексикою і негативним денотатом. Ця лінія проста й інтерпретована, але крихка до доменного зсуву та іронічних конвенцій спільнот.

- Глибокі нейромережі (окрім трансформерів). Огляд описує контекстні моделі, які поєднують локальний зміст твіта із ширшим контекстом (історія автора, попередні репліки).

- Трансформери. Найсильніші результати у спільних завданнях дала лінія, що поєднує BERT-подібні представлення з агрегацією контексту.

Додатково відзначаються ACE-моделі (Affective & Contextual Embeddings), які збагачують BERT афективними сигналами (емоційні лексики, багатоголова увага) і тим самим поліпшують розрізнення випадків із тонкими емоційними зсувами. Ключова теза [28]: сам по собі твіт часто недостатній -

потрібні діалогові нитки та історія автора. Це справедливо і для класичних лексичних рис (де контекст знижує кількість хибних спрацьовувань), і для DNN/трансформерів (де контекст інтегрується або як окремий вхід, або через спеціальні аугментації).

В роботах [28, 37] визначено сильні та слабкі сторони означених підходів. Лексичні системи - легкі для інтерпретації й швидкі, однак чутливі до саркастичних конвенцій спільнот і перенесення між доменами. Глибинні моделі без трансформерів краще «розуміють» локальні патерни й дають приріст від авторського/діалогового контексту, але потребують ретельної інженерії входів. Трансформери стабільно виграють, коли контекст введено правильно (через архітектуру або аугментацію) і коли зменшено шум у розмітці. Загальні болі галузі: нестандартизовані хештеги й емодзі, як слабкі проксі розмітки, суб'єктивність анотації, а також відтворюваність експериментів через «висихання» твітів при гідратації ID (частина публічних твітів згодом зникає). Усі ці фактори спричиняють розбіжності в оцінках і ускладнюють чесне порівняння методів.

Практичний висновок для емоційного аналізу. Сарказм - «інвертор полярності», тож пропуск цього шару систематично псує сентимент-оцінку. Найкращі результати нині дають трансформери, що явно моделюють діалоговий/авторський контекст або поєднують контекст з афективними ознаками. У конвеєрах розпізнавання емоційної тональності це варто оформлювати, як попередній крок обробки тексту (детектор сарказму) або як багатозадачне навчання з доступом до контексту, мінімізуючи шум у розмітці й використовуючи аугментацію діалогами.

У статті [33] автори зосереджуються на задачі адаптації між доменами в аналізі емоційної тональності - складній проблемі, коли модель, навчена на одному джерелі даних (наприклад, відгуках про книги), має коректно працювати на іншому (приклад, відгуках про електроніку). Вони використовують загальнодоступний набір даних Amazon Reviews, який містить багатодоменні підкорпуси, що дозволяє тестувати як однодоменну

адаптацію (англ. single-source domain adaptation), так і багатодоменну адаптацію (multi-source domain adaptation). Автори підкреслюють, що класичні методи, які вчать окрему модель для кожного домену, виявляються неефективними: значна частина параметрів дублює знання про базову структуру мови, тому новий підхід має забезпечити спільне навчання інваріантних представлень.

Ключовою новацією є Adversarial Soft Prompt Tuning (AdSPT) - стратегія, яка навчає м'які підказки (англ. soft prompts) для токена «[MASK]», не змінюючи параметри самої мовної моделі. Ці підказки виступають як диференційовні вектори, що вставляються у вхідну послідовність перед текстом, і саме вони кодують специфіку домену. Механізм змагального навчання (англ. adversarial training) вводиться через дискримінатор домену, який намагається розрізнити, з якого джерела походить приклад, тоді як генератор (основна модель із soft prompt) - навпаки, прагне зробити свої представлення настільки універсальними, щоб дискримінатор не зміг відрізнити домен. Таким чином, модель вивчає домен-інваріантні знання для токена «[MASK]», що дозволяє переносити емоційно-тональні закономірності між різними типами текстів.

Архітектура AdSPT ґрунтується на вже натренованих попередньо-навчених мовних моделях (PLM) - таких як BERT та GPT, - які виступають як основа для побудови представлень. Soft prompt tuning дозволяє зберегти переваги цих великих моделей, не потребуючи повного донавчання, що зменшує витрати на обчислення та обсяг даних. У результаті модель демонструє кращу узагальнювальну здатність на доменах, які не були представлені під час тренування, та перевершує базові методи на Amazon Reviews за метриками точності й макро-F1. Дослідники також показують, що підхід AdSPT стійкий до «шуму» у даних і стабільно працює як у випадку одиночного джерела, так і при комбінації кількох доменів, що робить його придатним для практичних застосувань у багатодоменних системах аналізу емоційної тональності.

Методи, використані в статтях [37, 38], включають позначення послідовності слів та класифікацію речень. Модель маркування послідовності слів приймає послідовність слів, як вхідні дані та призначає кожному слову мітку. З іншого боку, модель класифікації речень бере послідовність речень і класифікує кожне речення як таке, що містить стимул чи ні. Модель для спільної класифікації речень дещо відрізняється від попередніх моделей, оскільки робить передбачення для речення у контексті всього тексту. Набори даних, які використовуються в експериментах, як власні, так і загальнодоступні. Вони включають набір даних EmotionStimulus, і набір даних ElectoralTweets.

Стаття [29] представляє метод «Попереднє навчання з розширеними знаннями про настрої» (англ. Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis, SKER), який включає знання про емоційну тональність тексту шляхом самонавчання. Метод SKER інтегрує явні знання про тональність уже на етапі попереднього навчання моделі. Спершу з неанотованих текстів автоматично видобуваються три типи знань: емоційні слова, полярність слів та пари «аспект-емоційне слово» (для пар аспектом вважається найближчий іменник у вікні до трьох токенів). Виділені елементи «вирізаються» за допомогою sentiment masking: маскуються до двох пар «аспект-емоційне слово» одночасно, далі - немасковані емоційні слова, а за потреби додаються «звичайні» токени для балансу.

Модель навчають відновлювати вилучену інформацію за трьома цілями: SW (прогноз емоційного слова), WP (прогноз його полярності) та AP (прогноз складу пар «аспект-емоційне слово»). Для AP використовують мультиміткову класифікацію з опорою на представлення «[CLS]», щоб явно моделювати залежність усередині пари; саме це відрізняє SKER від стандартного незалежного передбачення токенів у BERT. Знання для видобутку формують нескладним PMI-методом із малого набору слів, тож додаткової ручної розмітки не потрібно. Результати підтверджують внесок кожної складової. До навчання на датасеті Amazon із випадковим маскуванням дає невеликий

приріст, але найбільший поштовх забезпечує саме SW; подальше додавання WP та AP дає стабільні покращення, зокрема на тонких завданнях (аспект-рівень і role labeling). Перехід від незалежного передбачення елементів пари (AP-I) до мультиміткової AP-цілі помітно допомагає саме там, де важлива коректна взаємодія «аспект-оцінка». Візуалізації уваги використаних моделей токену «[CLS]» показують кращу зосередженість SKER на релевантних емоційних словах та аспектах.

Методи, використані в дослідженні [32], включають моделі глибокого навчання для виявлення емоційної тональності тексту. Автори використовують глибоку нейронну мережу (ГНМ), що складається з входу, виходу та набору прихованих шарів із кількома вузлами. Процес навчання ГНМ складається з попереднього навчання та етапу донавчання. Дослідження використовує кілька наборів даних. Автори зібрали твіти користувачів, опубліковані на ранніх етапах кризи COVID-19. Вони також використовували загальнодоступний набір даних для навчання класифікатора оцінки полярності настроїв. Для навчання класифікатора емоцій використовувався набір даних Emotional Tweets. У статті також визнаються обмеження запропонованої моделі, такі як її нездатність зрозуміти контекст, особливо коли він є саркастичним, і її зосередженість на твітах лише англійською мовою.

У статті [40] також обговорюють використання нейронних мереж у своїх дослідженнях по розпізнаванню емоційної тональності у новинах. Згадується використання згорткових нейронних мереж і довгострокових мереж короткочасної пам'яті для аналізу настроїв коротких текстів. У дослідженні використовуються різноманітні набори даних, як створені власноруч, так і публічні. Автори зазначають, що перші моделі використовували власні спеціально створені набори даних для експериментів. В статті міститься комплексний огляд ролі розпізнавання емоційної тональності у виявленні фейкових новин, обговорюються різні методи та прийоми, які використовуються для включення тональності у процес виявлення фейкових новин.

В дослідженнях також можуть використовуватися комбінації моделей. У [30] задачу Aspect-Based Sentiment Analysis (ABSA) формалізовано як класифікацію пари “текст $\leftrightarrow$ аспект” у схемі sentence-pair: першим входом іде речення або відгук, другим - вербалізований опис категорії аспекту (автори перетворюють формати типу «ENTITY#ASPECT» на природні фрази на кшталт “food, style options”, аби краще узгодити їх із BERT). Виявлення категорії аспекту реалізують як двійкову класифікацію “related/unrelated” для кожної потенційної категорії, що дає змогу працювати й поза доменом (для аспектів, відсутніх у тренувальному наборі, але поданих текстом). Щоб сформувані негативні приклади, усі аспекти, не анотовані для даного тексту у SemEval-2016 Task 5, автоматично маркуються як unrelated; через сильний дисбаланс класів автори застосовують вагові коефіцієнти втрат, зменшуючи вагу частих міток.

Класифікатор полярності будується аналогічно - як sentence-pair модель, що повертає одну з міток {positive, negative, neutral, conflict} для заданої пари “текст–аспект”. Далі автори пропонують об’єднану модель, яка в єдиному багато-класовому виході повертає або одну з тональних міток (коли аспект релевантний), або мітку unrelated, тим самим знімаючи потребу у каскаді окремих моделей і зменшуючи акумуляцію помилок. Оцінювання проводиться на підзадачах SemEval-2016 (рівні речення й документ), включно з поза-доменним тестуванням на корпусі Hotel (для текстового рівня його зібрано шляхом конкатенації речень, бо оригінально він був лише реченнєвим). У більшості порівнянь combined-модель перевершує чистий тональний класифікатор.

Дискусія вказує на важливі нюанси. По-перше, аспектні моделі стабільно випереджають саме у задачі “related/unrelated”, адже модель мусить одночасно вирішувати і релевантність, і полярність, що підвищує складність. По-друге, sentence-level часто не несе достатньо інформації, щоб надійно вирішити релевантність аспекта; це шкодить переносимості, тоді як тренування на sentence-level краще поводить в поза-доменних сценаріях.

По-третє, змішане тренування на “both” (restaurants+laptops) не завжди допомагає через нерівномірність простору аспектів (наприклад, у «laptop» є значно більше унікальних категорій, що тягне падіння precision). Загалом автори пов’язують виграші з тим, що BERT і sentence-pair порівняння “аспект–текст” краще захоплюють семантичну подібність і контекст, що критично для ABSA

В результаті проведеного комплексного аналізу сучасних методів та підходів до вирішення задачі розпізнавання емоційної тональності тексту було сформовано набір ключових характеристик, що дозволяють систематизувати та порівняти існуючі рішення. Ці характеристики, наведені в табл. 1.1, відображають фундаментальні аспекти, що визначають архітектуру, функціональність та сферу застосування відповідних моделей і систем. Вибір цих критеріїв зумовлений необхідністю охоплення основних вимірів проблеми: від глибини емоційного аналізу (розпізнавання базових емоцій проти сентименту) до масштабу обробки текстових даних (рівень висловів, речень чи цілих документів).

В подальшому цей базовий перелік характеристик може бути модифікований та розширений з урахуванням новітніх досягнень, таких як поява мультимодальних моделей, що аналізують текст у поєднанні з іншими даними, та поглиблення деталізації задачі, наприклад, через розпізнавання інтенсивності емоцій або складних емоційних станів, як-от сарказм.

Таблиця 1.1

Перелік характеристик методів розпізнавання емоційної тональності
тексту

№	Опис характеристик
N ₁	Розпізнавання основних емоцій
N ₂	Розпізнавання лише сентиментів
N ₃	Розпізнавання на рівні окремих висловів
N ₄	Розпізнавання на рівні речень

Продовження таблиці 1.1

N ₅	Розпізнавання на рівні документу
N ₆	Використання лексичного аналізу
N ₇	Використання сучасних нейромережових засобів
N ₈	Використання відомих наборів даних
N ₉	Використання власних наборів даних
N ₁₀	Можливість врахування специфіки української мови
N ₁₁	Використання комбінації моделей для розпізнавання
N ₁₂	Використання однієї моделі для розпізнавання
N ₁₃	Точність розпізнавання
N ₁₄	Оперативність розпізнавання

Важливим аспектом класифікації є дихотомія між лексичними та нейромережевими підходами. Лексичний аналіз (N₆), що спирається на словники тональності (сентимент-лексикони), є одним із основних методів, проте його ефективність обмежується статичністю словника та нездатністю повною мірою враховувати контекстуальні залежності.

На противагу цьому, сучасні нейромережові засоби (НМЗ) (N₇), зокрема архітектури на основі трансформерів, демонструють значно вищу точність завдяки здатності вловлювати складні семантичні зв'язки у тексті. Використання стандартних (N₈) чи власних (N₉) наборів даних визначає відтворюваність та специфічність моделі, тоді як багатомовна підтримка (N₁₀) є критичною для універсальних комп'ютерних систем.

Нарешті, вибір між єдиною (N₁₂) та комбінованою (N₁₁) моделлю відображає компроміс між простотою імплементації та потенційним підвищенням точності за рахунок ансамблевих методів. Характеристики (N₁₃ та N₁₄) стосуються таких показників якості розпізнавання як точність та оперативність. Опис визначений характеристик наведено в табл. 1.1.

Проведений аналіз відомих методів дозволив використати вибрані характеристики для проведення порівняльної оцінки потенціалу розглянутих

підходів до розпізнавання емоційної тональності в тексті. Для забезпечення уніфікованого та наочного порівняння оцінки були призначені за допомогою бінарної шкали (0/1), що відображає відсутність або наявність відповідної функціональної можливості чи архітектурної особливості. Такий підхід, хоч і є певним спрощенням, дозволяє швидко ідентифікувати ключові переваги та недоліки кожного методу, а також виявити загальні тенденції у розвитку галузі.

Отримані оцінки характеристик для репрезентативної вибірки методів частково представлені в табл. 1.2. При цьому в табл. 1.2 визначені в першій колонці номери рішень відповідають:

- P1 - Technological Approaches to Detecting Online Disinformation and Manipulation,
- P2 - Effective Token Graph Modeling using a Novel Labeling Strategy for Structured Sentiment Analysis,
- P3 - A Survey on Automated Sarcasm Detection on Twitter,
- P4 - Adversarial Soft Prompt Tuning for Cross-Domain Sentiment Analysis,
- P5 - Token Sequence Labeling vs. Clause Classification for English Emotion Stimulus Detection,
- P6 - SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis,
- P7 - Cross-Cultural Polarity and Emotion Detection Using Sentiment Analysis and Deep Learning on COVID-19 Related Tweets,
- P8 - Sentiment Analysis for Fake News Detection,
- P9 - Aspect-Based Sentiment Analysis using BERT.

Таблиця 1.2

Оцінки характеристик методів розпізнавання

Номер рішення	Характеристики													
	N ₁	N ₂	N ₃	N ₄	N ₅	N ₆	N ₇	N ₈	N ₉	N ₁₀	N ₁₁	N ₁₂	N ₁₃	N ₁₄
P1	1	0	1	0	0	1	1	1	0	0	0	1	1	0
P2	0	1	1	0	1	0	1	1	0	0	0	1	1	0
P3	0	0	0	0	1	1	1	1	1	0	1	0	0	0
P4	0	1	0	0	1	0	1	1	0	0	0	1	1	0
P5	1	0	1	1	0	0	1	1	1	0	1	0	1	0
P6	0	1	0	0	1	0	1	1	0	0	0	1	1	0
P7	1	0	0	0	1	1	1	1	1	0	0	1	0	0
P8	0	1	0	0	1	1	1	1	1	0	0	1	0	0
P9	0	1	1	0	1	0	1	1	0	0	1	0	1	0

Варто зазначити, що до таблиці порівняльного аналізу не були включені характеристики, що є спільними та притаманними для всіх сучасних підходів. До таких універсальних ознак належать, наприклад, етапи попередньої обробки тексту (тобто, токенізація, лематизація, видалення стоп-слів), які є стандартною практикою незалежно від обраної моделі. Включення подібних загальних характеристик до порівняльної таблиці не мало б диференціюючої цінності та лише ускладнило б аналіз, не додаючи інформативності щодо унікальних властивостей кожного з розглянутих методів. Таким чином, фокус аналізу було свідомо зміщено на ті аспекти, що дозволяють виявити принципові відмінності між архітектурами та методологіями.

Виходячи з аналізу табл. 1.2 майже всі проаналізовані методи мають певні недоліки, так як, в більшості концентруються на знаходження сентименту, тобто тільки три емоції (негатив, позитив та нейтрал), на рівні документа. Також вони погано пристосовані до більш складніших емоції та до розпізнавання на рівні речень, окремих висловів. Формування переліку

характеристик розпізнавання емоційної тональності в тексті дозволяє окреслити напрямок наступного етапу досліджень – розробкою математичного забезпечення процедури оцінки ефективності засобів розпізнавання та інтеграції засобів розпізнавання емоцій на основі аналізу текстових фрагментів. Крім іншого, це забезпечить розробку методу та моделей розпізнавання емоційної тональності в тексті, з покриттям основних емоцій на рівні речень та окремих висловів. Базуючись на підходах, що викладенні у роботах [28, 41, 42], доцільно використовувати комбінацію моделей та відомі набори даних, з можливістю розширення власними даними, для підвищення точності та повноти розпізнавання. Також базуючись на результатах [34, 43, 44] можна вважати доцільним розробку методів інтеграції засобів розпізнавання емоцій на основі аналізу текстових фрагментів та голосових повідомлень користувачів КС. Одночасно, в доступній літературі, що була проаналізована, не наведено обґрунтованого переліку характеристик засобів РЕТ, які повинні відповідати вимогам поставленої задачі. Притаманність таких характеристик і їх належна адаптація до поставленої задачі визначення емоційного забарвлення тексту відіграють важливу роль в ефективності застосування цих методів.

1.3. Оцінка перспектив підвищення ефективності розпізнавання емоційного забарвлення фрагментів текстів українською мовою

Аналіз, проведений у попередніх п. 1.1, 1.2, демонструє значний прогрес у розробці моделей РЕТ, що зумовлений передусім домінуванням архітектур глибокого навчання, зокрема трансформерів [46-47] . Водночас цей прогрес висвітлює дві системні проблеми, що обмежують практичне впровадження таких рішень.

По-перше, більшість сучасних досліджень має виражений “моделецентричний” характер. Основна увага в них приділяється інкрементальному покращенню метрик якості (таких як точність, повнота, F1-міра) на еталонних наборах даних, часто ігноруючи обчислювальну вартість,

енергоефективність та можливість розгортання моделей поза межами високопродуктивних серверних середовищ [3]. Це створює значний розрив між академічними досягненнями та практичною імплементацією в універсальних КС, які охоплюють широкий спектр пристроїв - від потужних хмарних кластерів до мобільних телефонів та вбудованих систем.

По-друге, для української мови, яка належить до категорії низькоресурсних (low-resource languages) у галузі обробки природної мови [6, 22], виникає додатковий комплекс викликів. Пряме застосування багатомовних моделей, попередньо навчених на масивах даних, де домінує англійська мова, часто не враховує унікальних лексичних, синтаксичних та культурних особливостей українського мовного простору. Це особливо критично для розпізнавання складних емоційних станів, таких як іронія та сарказм, що глибоко залежать від соціокультурного контексту [48]. Існуючий дефіцит великих, якісно анотованих та збалансованих україномовних корпусів текстів для аналізу тональності суттєво обмежує можливості навчання ефективних моделей та їх адекватну валідацію [49]. Традиційний ручний процес анотування є надзвичайно трудомістким, дорогим та повільним. У цьому аспекті перспективним напрямом є залучення сучасних великих мовних моделей (LLM) для значного прискорення процесу розмітки. Такий підхід, відомий як “human-in-the-loop” (людина-в-контурі), передбачає використання LLM не як повної заміни експерта, а як потужного інструмента для “холодного старту”. Модель у режимі “zero-shot” або “few-shot” генерує початкові мітки емоційної тональності для значного масиву нерозмічених даних, після чого ці анотації верифікуються та коригуються людиною-анотатором. Це значно знижує когнітивне навантаження на експертів і в разі пришвидшує процес. Дослідження підтверджують, що для багатьох завдань текстової класифікації сучасні LLM досягають якості, співмірної або навіть вищої за якості неекспертних анотаторів [50], що робить їх ідеальним інструментом для швидкого створення великих та якісних україномовних датасетів. Таким чином, розробка перспективного рішення вимагає комплексного підходу, що

адресує як проблему обчислювальної ефективності, так і проблему браку лінгвістичних ресурсів.

Одним із актуальних напрямків дослідження полягає у здійсненні зсуву від розробки ізольованих, монолітних моделей до створення комплексного методу, що охоплює весь життєвий цикл проектування, розгортання та супроводу системи розпізнавання емоційної тональності. Такий перехід від “моделецентричного” до “системоцентричного” підходу вимагає відповідної еволюції критеріїв оцінки. Традиційні метрики якості, хоча й залишаються важливими, є недостатніми для комплексної оцінки. Тому пропонується, щоб розроблений метод супроводжувався багатокритеріальною системою оцінки, яка б дозволяла проводити всебічний аналіз на ключових рівнях. Перші два рівні, що стосуються точності та оперативності розпізнавання, формують фундамент оцінки. На рівні якості використовуються класичні метрики (точність, повнота, F1-міра), що підтверджують базову спроможність моделі вирішувати поставлену задачу. Однак для універсальних комп’ютерних систем цього недостатньо, тому на рівні ефективності оцінюються нефункціональні характеристики, що безпосередньо впливають на можливість практичного розгортання: швидкість інференсу на цільових апаратних профілях (CPU, GPU, мобільні процесори), пікове споживання оперативної пам’яті (англ. Random Access Memory, RAM), фізичний розмір моделі та кількість її параметрів, а також пропускна здатність системи. Окремим імперативом є оцінка енергоспоживання та його вуглецевого сліду (CO₂-еквівалент) відповідно до протоколів Green AI, що стає дедалі важливішим у контексті сталого розвитку обчислювальних технологій [51]. Третій, не менш важливий рівень оцінки - надійність та узгодженість - для побудови довіри до системи та її відповідального впровадження. Цей рівень охоплює стійкість моделі до нетипових вхідних даних, таких як лінгвістичний шум (описки, сленг), суржик та код-світчинг (змішування мов), що є поширеним явищем в українському інтернет-просторі. Крім того, до цього рівня належить простежуваність рішень, дотримання правил, регламентів, специфікацій та відтворюваність

експериментів, що є фундаментальними для наукової валідності та подальшого вдосконалення методу. Критичним компонентом є аналіз та контроль упереджень, закладених у навчальних даних, які можуть призводити до систематичних помилок та дискримінаційних рішень щодо певних соціальних груп. Нехтування цим аспектом може перетворити навіть технічно досконалу модель на інструмент посилення суспільної нерівності, що є неприпустимим для систем штучного інтелекту, призначених для широкого використання [52].

Розроблення методу та моделей має формалізувати послідовність етапів, починаючи з аналізу вимог та обмежень цільової платформи, і завершуючи розгортанням оптимізованого рішення. Вона повинна системно поєднувати: підготовку даних, їх обробку, вибір архітектурних рішень, методів навчання та технік подальшої оптимізації. Кінцевою метою є створення не однієї універсальної моделі, а гнучкого методу, що дозволяє створювати ціле сімейство моделей, кожна з яких оптимізована для конкретного сегмента універсальних комп'ютерних систем. Таким чином, метою даного дисертаційного дослідження є розробка моделей та методу розпізнавання емоційної тональності фрагментів тексту, для української мови що базується на формалізованому процесі їх створення. Цей метод має бути адаптованим до лінгвістичних реалій українського простору та забезпечувати створення обчислювально ефективних моделей. На цьому ґрунтується актуальність науково-прикладної проблеми розробки такого комплексного рішення.

1.4. Висновки до першого розділу

У першому розділі дисертаційної роботи проведено комплексний аналіз предметної області розпізнавання емоційної тональності тексту в контексті універсальних комп'ютерних систем. Встановлено, що ця задача набуває виняткової актуальності через експоненціальне зростання обсягів текстових даних та її широкий прикладний потенціал. Для української мови ця проблематика загострюється низкою специфічних викликів: статусом низькоресурсної мови, що зумовлює дефіцит якісних анотованих корпусів;

лінгвістичними особливостями (складна морфологія, суржик, код-світчинг); та стратегічною важливістю в умовах протидії ворожим інформаційно-психологічним операціям. Водночас виявлено фундаментальну науково-прикладну проблему: існує суттєвий розрив між високою точністю сучасних моделей глибокого навчання, зокрема трансформерів, та їхньою практичною придатністю для розгортання в гетерогенних обчислювальних середовищах через надмірні вимоги до апаратних ресурсів.

Систематизація та аналіз відомих засобів розпізнавання емоційної тональності, підсумований у табл. 1.1 та 1.2, було обґрунтовано базовий перелік характеристик засобів для автоматичного розпізнавання емоційного забарвлення в тексті. Цей перелік дозволяє оцінювати, наскільки обрані методи відповідають вимогам задачі розпізнавання емоційного тону тексту. Більшість досліджень концентрується на вирішенні вузьких завдань, таких як класифікація полярності на рівні документа, ігноруючи потребу в більш гранульованому аналізі дискретних емоцій на рівні окремих висловів та речень. Було виявлено відсутність єдиної методології оцінки, яка б системно поєднувала вимоги до якості розпізнавання з критеріями обчислювальної ефективності, надійності та етичності. Таким чином, проведений аналіз обґрунтовує потребу в переході до створення модульної конструкції системи розпізнавання. В підсумку в результаті проведеного аналізу визначено, що важливим напрямком підвищення ефективності розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах є вдосконалення нейромережових засобів розпізнавання. Для цього необхідно розвинути методологічну базу, розробити метод розпізнавання емоційної тональності та створити анотований корпус україномовних текстів, які забезпечать їх адаптацію до застосування в умовах обмежених обчислювальних ресурсів з урахуванням лінгвістичної специфіки української мови.

РОЗДІЛ 2

ЕЛЕМЕНТИ МЕТОДОЛОГІЧНОГО ЗАБЕЗПЕЧЕННЯ РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ

2.1. Концептуальна модель розпізнавання емоційної тональності текстових фрагментів

Проведений у попередньому розділі аналіз виявив низку системних недоліків існуючих підходів до розпізнавання емоційної тональності. Це обґрунтовує необхідність переходу від розробки ізольованих рішень до створення комплексного методу. Першим і фундаментальним кроком на шляху формалізації такого методу є розробка його концептуальної моделі. Саме концептуальна модель покликана визначити ключові компоненти процесу розпізнавання емоційної тональності, встановити взаємозв'язки між ними та описати послідовність етапів, починаючи від збору та підготовки даних і закінчуючи використанням та оцінкою моделі.

Відсутність формалізованого, узагальненого опису процесу розпізнавання емоційного забарвлення тексту призводить до невизначеності та ускладнює розробку дійсно ефективних засобів [4]. Різноманіття підходів, що варіюються від використання лексичних ознак та мультиміткової класифікації до застосування складних архітектур глибокого навчання, таких як згорткові нейронні мережі (англ. Convolutional Neural Network, CNN), рекурентні нейронні мережі (англ. Recurrent Neural Network, RNN), трансформер (англ. Transformer) та їх комбінацій, підкреслює відсутність єдиної, системної рамки. Як зазначається у сучасних оглядах, дослідники часто змушені емпірично комбінувати різні етапи, що ускладнює відтворюваність результатів та порівняльний аналіз. Для подолання вказаного недоліку та структурування процесу розробки пропонується розробити концептуальну модель процесу розпізнавання емоцій, що слугуватиме теоретичним фундаментом для побудови цілісного методу.

Розробка такої концептуальної моделі є важливою не лише з теоретичної,

а й з практичної точки зору, оскільки вона може слугувати основою для створення більш ефективних та універсальних алгоритмів розпізнавання емоцій у текстах. Це матиме безпосереднє застосування в таких галузях, як аналіз соціальних мереж, моніторинг клієнтського досвіду та маркетинг. Формалізована модель дозволить систематизувати існуючі підходи, виявити їхні переваги та недоліки, а також допоможе визначити пріоритетні напрямки для подальших наукових досліджень. Це, у свою чергу, сприятиме глибшому розумінню природи емоцій у текстових даних, розробці нових методів їх представлення та вилучення, а також удосконаленню архітектур нейронних мереж для обробки емоційної інформації. Крім того, концептуальна модель може стати основою для створення уніфікованих україномовних емоційних корпусів та стандартизованих багатокритеріальних метрик оцінювання, що значно полегшить валідацію та об'єктивне порівняння різних моделей і алгоритмів між собою.

У загальному випадку концептуальна модель розпізнавання емоційного забарвлення та тональності у текстах являє собою модель предметної області, що складається з переліку взаємопов'язаних концепцій та понять, які використовуються для опису цієї області. Вона включає властивості та характеристики емоційних проявів у текстах, класифікацію емоцій за типами, ситуаціями, ознаками, а також приховані патерни та принципи, що керують емоційними процесами у текстових даних. Концептуальна модель, в даному випадку, є відображенням концепції емоційності тексту, під якою розуміється спосіб трактування емоційних проявів, основна точка зору та провідні ідеї для їх систематичного аналізу та інтерпретації.

Для визначення ефективності процесу РЕТ тексту доцільно використовувати визначення та підходи з галузі комп'ютерної лінгвістики, обробки природної мови та машинного навчання, оскільки ці галузі лежать в основі методів та моделей для автоматизованого виявлення емоцій у текстових даних. Концептуальна модель має охоплювати різні аспекти емоційності текстів та забезпечувати систематичний підхід до їх аналізу та інтерпретації.

Згідно міжнародних стандартів, ефективність програмної системи визначається низкою атрибутів, що характеризують її продуктивність, споживання ресурсів та дотримання встановлених норм і вимог. Ключовими показниками ефективності є швидкість виконання операцій, обсяг спожитих обчислювальних ресурсів та відповідність системи галузевим стандартам. Зокрема, до основних атрибутів ефективності належать: швидкість (час відгуку та обробки даних), ресурсоемність (споживання апаратних та програмних ресурсів) і узгодженість (дотримання правил, регламентів та специфікацій).

На першому етапі створення концептуальної моделі для РЕТ у текстах було проведено гармонізацію термінології, що використовується в цій галузі. Необхідність цього кроку зумовлена тим, що предметна область знаходиться на перетині комп'ютерної лінгвістики, психології та машинного навчання, що призводить до існування різних, іноді суперечливих, трактувань ключових понять. Гармонізація виконувалась з метою відобразити сучасний стан наукових досліджень, забезпечити концептуальну ясність та створити однозначний семантичний простір для подальшої формалізації. В результаті було визначено та деталізовано такий набір ключових термінів.

До групи предметної області належать поняття, що описують об'єкт та предмет дослідження - емоції та їх прояви у текстових даних:

- Базові емоції - фундаментальні, психологічно неподільні емоційні стани, що вважаються універсальними для різних культур. У даній роботі, спираючись на класичну модель Пола Екмана, до базових емоцій належать гнів, огида, смуток, зневага, страх, подив та радість. Ця дискретна модель є основою для задач багатокласової класифікації емоцій.

- Емоції в тексті - це суб'єктивне емоційне забарвлення, яке виражається автором через лексичні, синтаксичні та прагматичні засоби мови і передає певні почуття, настрої або ставлення до описуваних явищ. Важливо розрізняти інтенцію автора (емоція, яку він мав намір виразити) та перцепцію читача (емоція, яка сприймається), оскільки вони можуть не збігатися. У

рамках даного дослідження розпізнавання емоцій фокусується на ідентифікації маркерів, що з високою ймовірністю вказують на авторську інтенцію.

- Фрагмент тексту - обмежена послідовність токенів $T = (w_1, w_2, \dots, w_n)$, що формує відносну змістовну цілісність. Таким фрагментом може виступати словосполучення, речення або група речень (наприклад, абзац чи коментар у соціальній мережі).

- Емоційні ознаки - лінгвістичні, стилістичні або контекстуальні маркери в тексті, що корелюють з певними емоціями. До них належать емотивно забарвлена лексика, інтенсифікатори (напр., "дуже", "неймовірно"), модальні дієслова, специфічні синтаксичні конструкції (оклики, риторичні питання), а також паралінгвістичні засоби (емодзі, використання великих літер, знаки пунктуації).

- Тональність (англ. Sentiment) - загальна емоційна полярність фрагмента тексту, що зазвичай класифікується як позитивна, негативна або нейтральна. Для більш гранульованого аналізу тональність може бути представлена, як неперервна величина. Формально, це функція $S(T)$, що відображає текстовий фрагмент у числовий проміжок:

$$S(T) \rightarrow [-1, 1], \quad (2.1)$$

де $S(T) = 1$ відповідає максимально позитивній тональності, $S(T) = -1$ – максимально негативній, а $S(T) = 0$ – нейтральній.

Емоційне забарвлення - це багатовимірна міра присутності базових емоцій у текстовому фрагменті. На відміну від одновимірної тональності, емоційне забарвлення дозволяє фіксувати складні емоційні стани (наприклад, одночасну присутність радості та подиву). Воно може бути представлене у вигляді вектора $E(T)$, де кожен елемент відповідає інтенсивності однієї з k базових емоцій:

$$E(T) = (e_1, e_2, \dots, e_k), \quad (2.2)$$

де e_i – інтенсивність i -ї базової емоції, 0 – означає її повну відсутність, а 1 – максимальну присутність.

До групи термінів методології, що стосується інструментарію та використовується для розв'язання задачі розпізнавання емоцій належать наступні терміни:

- Нейронна мережа - обчислювальна система, що імітує структуру та функціонування біологічних нейронних мереж. Вона складається з взаємопов'язаних простих обчислювальних елементів (штучних нейронів), організованих у шари.

- Глибинне навчання - підрозділ машинного навчання, що використовує ГНМ, тобто нейронні мережі з великою кількістю прихованих шарів між вхідним та вихідним. Ключова перевага глибинного навчання полягає у здатності автоматично вивчати ієрархічні представлення даних: нижні шари мережі виявляють прості ознаки (напр., n -грами слів), а верхні комбінують їх для виявлення більш складних, абстрактних патернів (семантичних та емоційних конструкцій).

- Нейромережева модель (НММ) - це конкретний екземпляр нейронної мережі з визначеною архітектурою та навченими ваговими коефіцієнтами, призначений для розв'язання певної задачі (в даному випадку - розпізнавання емоційної тональності).

До термінів оцінки, що визначають критерії для вимірювання якості та надійності розроблених моделей, входять наступні визначення похибок:

- Похибка першого роду – помилка, яка виникає, коли система невірно ідентифікує наявність емоції (або позитивної тональності) у фрагменті, який насправді є нейтральним або має іншу емоцію. Це відповідає ситуації, коли нейтральний приклад класифікується як емоційно забарвлений.

- Похибка другого роду – помилка, яка виникає, коли система не розпізнає наявну емоцію (або тональність) у фрагменті, класифікуючи його як нейтральний. Це відповідає пропуску релевантного емоційного сигналу.

- Якість розпізнавання - це інтегральна характеристика процесу функціонування системи в умовах ресурсних обмежень, яка визначається як синергетичне поєднання точності розпізнавання (ступенем збігу результатів роботи моделі з еталонними мітками) та оперативності розпізнавання (часовою ефективністю обробки запитів). В рамках даного дослідження висока якість розпізнавання досягається лише за умови одночасного забезпечення необхідного рівня точності та дотримання вимог щодо допустимої латентності для конкретної універсальної комп'ютерної платформи

Ці визначення, узгоджені з сучасними поглядами на природу емоцій та їх прояв у текстових даних, створюють концептуальну основу для формального опису та моделювання процесу виявлення емоційного забарвлення фрагментів тексту. Вони закладають фундамент для розробки уніфікованої концептуальної моделі, яка буде представлена в наступних підрозділах.

Окрім того, в рамках дослідження цієї статті було встановлено, що концептуальна модель є засобом для систематизації причинно-наслідкових відносин, які є характерними для процесу ідентифікації емоцій. Це обумовлено потребою в оптимізації ефективності вказаного процесу.

Розробка концептуальної моделі для розпізнавання емоційної тональності та забарвлення у текстах вимагає системного підходу, що враховує кілька ключових аспектів.

По-перше, необхідно чітко визначити умови функціонування системи розпізнавання: цільові апаратні платформи, вимоги до швидкодії та ресурсів, а також її взаємодію з іншими програмними компонентами.

По-друге, потрібно забезпечити методологічну основу для ефективного застосування НММ, що охоплює весь її життєвий цикл від підготовки даних до навчання та валідації.

По-третє, модель повинна передбачати можливості управління системою та визначення її керованих параметрів, що дозволить адаптувати її до конкретних прикладних завдань.

Згідно із загальноприйнятою технологією використання НММ та враховуючи виявлені в першому розділі проблеми, процес розпізнавання емоційної тональності у текстах доцільно декомпонувати на основні, логічно пов'язані кроки. Ця послідовність кроків зображена на рис. 2.1.

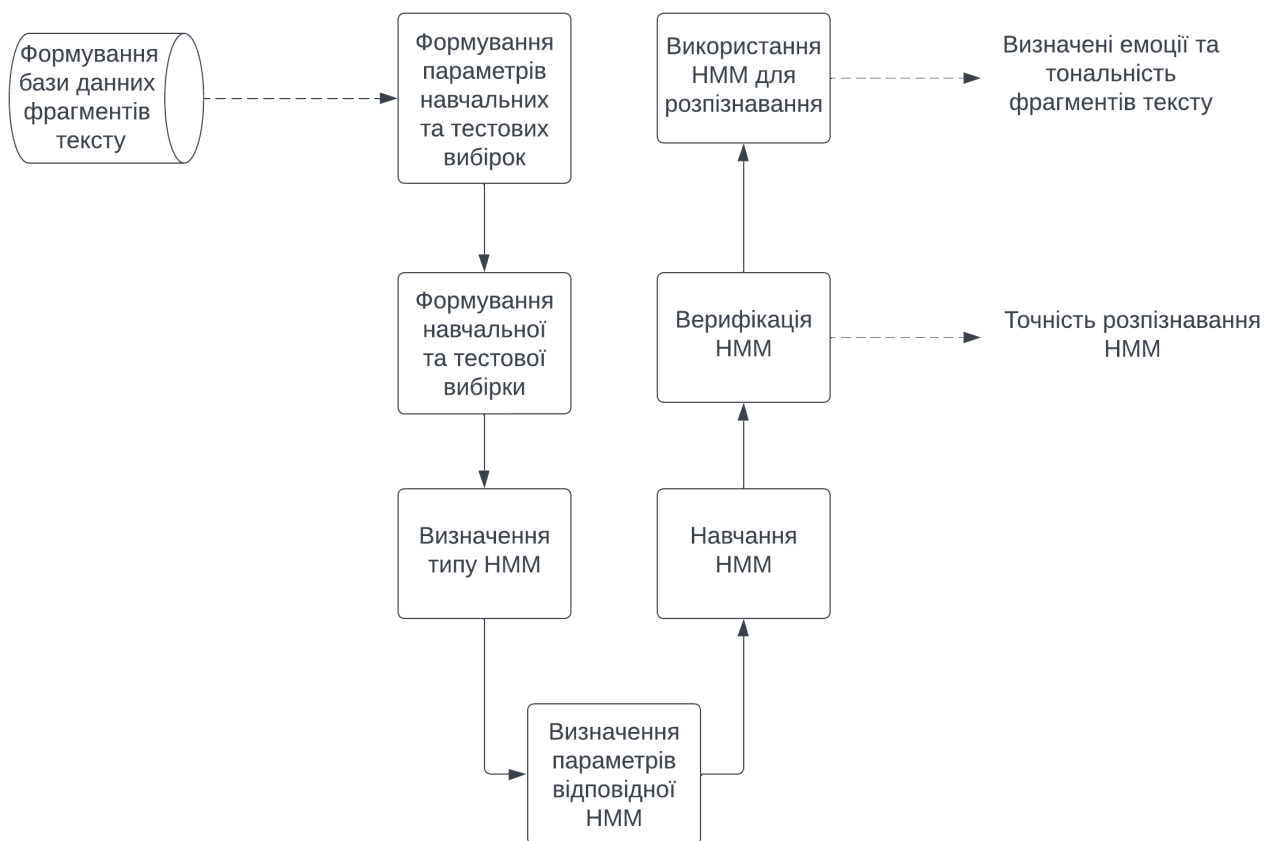


Рис. 2.1. Схема застосування нейромережевих моделей для розпізнавання.

Така структура дозволяє формалізувати та систематизувати весь процес, роблячи його більш прозорим, керованим та відтворюваним.

Процес складається з таких кроків:

1. Формування бази даних фрагментів текстів. На цьому кроці здійснюється збір текстових даних із релевантних джерел (соціальні мережі, новинні портали, форуми відгуків), що відображають специфіку українського мовного простору. Далі відбувається процес анотування (розмітки) - присвоєння кожному текстовому фрагменту міток відповідних базових емоцій та/або тональності. Через високу вартість ручної розмітки, на цьому етапі

доцільно застосовувати підхід "людина-в-контурі" (human-in-the-loop), використовуючи великі мовні моделі (LLM) для генерації початкових анотацій з подальшою верифікацією експертами [53]. Результатом є створення високоякісної бази даних, що слугуватиме "грунтовою істиною" (ground truth) для подальших кроків.

2. Формування параметрів навчальних та тестових вибірок. На цьому кроці визначаються ключові параметри стратегії препроцесингу для формування вибірок, зокрема співвідношення для поділу даних на навчальну, валідаційну та тестову частини (напр., 80/10/10). Одночасно розробляється конвеєр попередньої обробки тексту, який може включати токенизацію, лематизацію (приведення слів до базової форми), видалення стоп-слів, нормалізацію (виправлення друкарських помилок, обробка сленгу та суржику). Цей крок є критично важливим для зменшення розрідженості даних та підвищення здатності моделі до узагальнення.

3. Формування навчальної та тестової вибірки. Підготовлені дані поділяються на навчальну (для тренування моделі), валідаційну (для налаштування гіперпараметрів та запобігання перенавчанню) і тестову (для фінальної, об'єктивної оцінки якості) вибірки. Після цього текстові фрагменти перетворюються у числовий формат, придатний для обробки нейронною мережею. Цей процес, відомий як *векторизація*, може використовувати різні методи - від класичних (TF-IDF) до сучасних контекстуалізованих векторних представлень (англ. embeddings) на основі трансформерних архітектур (напр., BERT).

4. Визначення типу НММ. На основі аналізу задачі, характеристик даних та вимог до обчислювальної ефективності обирається найбільш відповідна архітектура нейронної мережі. Це можуть бути згорткові нейронні мережі (CNN) для виявлення локальних ознак, рекурентні мережі (LSTM, GRU) для обробки послідовностей або трансформери для моделювання довготривалих залежностей у тексті. Визначаються керовані параметри моделі

(гіперпараметри): кількість шарів, кількість нейронів у шарах, швидкість навчання, розмір міні-вибірки тощо.

5. Навчання НММ. Обрана НММ навчається на навчальній вибірці шляхом ітеративної мінімізації функції втрат (напр., перехресної ентропії для задач класифікації) за допомогою алгоритмів градієнтної оптимізації (напр., Adam [54]). Процес навчання контролюється на валідаційній вибірці для відстеження перенавчання та своєчасного його припинення (early stopping).

6. Верифікація НММ та оцінка якості. Навчена НММ проходить всебічне тестування на відкладеній тестовій вибірці. Оцінка проводиться за комплексом метрик, що включають: метрики якості (точність, повнота, F1-міра, аналіз матриці плутанини для оцінки похибок 1-го та 2-го роду), метрики ефективності (час інференсу, споживання пам'яті, розмір моделі) та метрики надійності (стійкість до зашумлених даних). Результати оцінки порівнюються з наперед визначеними вимогами до точності та ресурсоемності.

7. Використання НММ для розпізнавання. Після успішної верифікації модель інтегрується у цільову програмну систему та використовується для розпізнавання емоційної тональності на нових, реальних текстових даних. Важливою складовою цього етапу є організація моніторингу продуктивності моделі в реальних умовах для виявлення деградації її якості з часом та планування періодичного перенавчання.

Представлена вище декомпозиція процесу розпізнавання емоцій на послідовні етапи створює структурований каркас для розробки. Однак практична реалізація цієї концептуальної моделі вимагає глибокого аналізу та вирішення низки складних науково-технічних завдань, що виникають на її ключових кроках застосування нейромережевої моделі. Хоча кожен крок має свою важливість, два з них є критичними, оскільки саме вони визначають фундаментальні властивості кінцевої системи: якість підготовлених даних та вибір архітектури нейромережевої моделі.

Перший крок, що стосується підготовчого етапу, полягає в обґрунтованому формуванні корпусу текстових даних. Це завдання виходить далеко за межі простого збору текстів.

Відповідно до принципу "сміття на вході - сміття на виході" (англю «Garbage In, Garbage Out»), якість та репрезентативність навчальної вибірки встановлюють верхню межу потенційної точності будь-якої, навіть найскладнішої, моделі. Корпус має адекватно відображати лінгвістичну реальність цільової області застосування - українського сегмента інтернету з усіма його особливостями: наявністю суржику, код-світчингу, неформальної лексики, сарказму та специфічних маніпулятивних прийомів. Незбалансованість класів емоцій або недостатнє представлення певних стилістичних явищ у навчальних даних неминуче призведе до створення упередженої моделі, яка буде демонструвати низьку стійкість та робити систематичні помилки в реальних умовах експлуатації.

Другий крок застосування, пов'язаний з етапом моделювання, полягає у визначенні оптимального виду та параметрів нейромережевої моделі. Це завдання є не просто технічним налаштуванням, а комплексною задачею багатоцільової оптимізації. Вибір архітектури (наприклад, між легковаговою CNN та потужним трансформером), її глибини, кількості параметрів та типу функції втрат безпосередньо визначає компроміс між предиктивною точністю моделі та її обчислювальною вартістю.

Надто складна модель може досягти високої точності на тестовій вибірці, але виявиться непридатною для розгортання на пристроях з обмеженими ресурсами (наприклад, мобільних телефонах) через високу затримку та споживання енергії. Навпаки, надто проста модель буде швидкою, але може виявитися нездатною вловити складні семантичні та емоційні залежності в тексті. Отже, вибір та конфігурація НММ у рамках запропонованої концептуальної моделі має розглядатися як керований процес пошуку оптимального балансу, що відповідає конкретним вимогам цільової універсальної КС.

Також слід зазначити для забезпечення репрезентативності даних та вибір оптимальної архітектури моделі - підкреслюють недосконалість повністю автоматизованих способів формування навчальних прикладів для нейромережових моделей, призначених розпізнавання емоційної тональності та забарвлення.

Проблема полягає в тому, що емоція в тексті є складним, контекстно-залежним та часто суб'єктивним явищем. Автоматичний збір даних не може гарантувати ані семантичного різноманіття, ані правильного балансу емоційних категорій. Більше того, встановлення релевантних емоційних міток для кожного прикладу - це нетривіальна задача, що стикається з проблемами багатозначності, іронії та сарказму, які не можуть бути надійно розпізнані без глибокого розуміння прагматики та культурного контексту.

Через ці фундаментальні особливості, запропонована концептуальна модель свідомо передбачає активне залучення експертів на ключових, початкових етапах процесу. Під експертом розуміється фахівець який володіє глибокими знаннями про способи вербального вираження емоцій в українській мові. Саме експертний досвід дозволяє перейти від механічного збору даних до цілеспрямованого формування якісного корпусу. Ця роль не обмежується лише анотуванням; вона включає визначення самих емоцій (наприклад, вибір базових емоцій та їх критеріїв), розробку інструкцій для анотаторів, що мінімізують суб'єктивність, та валідацію розмічених даних.

Таким чином, експерт виступає гарантом якості та адекватності вхідних даних, що є необхідною умовою для побудови надійної системи розпізнавання.

Як видно зі схеми зображеної на рис 2.2, «Експерт» є вихідною точкою всього процесу. Він формує три ключові артефакти:

- Емоційні параметри - концептуальний каркас задачі, що включає визначення набору базових емоцій та шкали тональності.
- Базу даних фрагментів текстів та Базу даних істинних прикладів, що слугує "ґрунтовою істиною" для навчання та тестування.

- Параметри НММ - початкові гіпотези та обмеження щодо архітектури моделі, засновані на аналізі складності підготовлених даних.

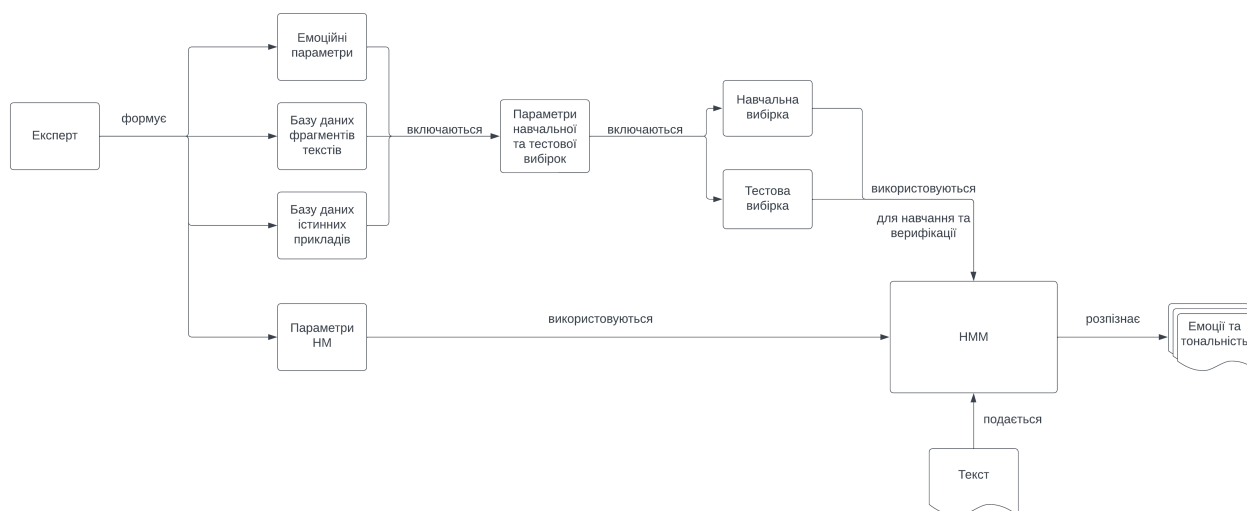


Рис. 2.2 Схема взаємодії компонентів НММ РЕТ у фрагментах текстів

Для подальшої деталізації необхідно проаналізувати ключові чинники, що безпосередньо впливають на ефективність нейромережевої моделі як центрального елемента цієї системи. Схема, зображена на рис. 2.3, декомпозує шлях до досягнення високої ефективності на три взаємопов'язані макропроцеси: обробку текстових фрагментів, проектування НММ та навчання НММ. Глибокий аналіз цих процесів та залежностей між ними дозволяє перейти від загальної концепції до конкретної методології розробки.

Належна обробка вхідних текстових даних є критично важливим попереднім етапом, що закладає фундамент для коректного навчання та функціонування НММ. Ефективність цього макропроцесу визначається якістю виконання низки послідовних кроків. Вибір мови та алфавіту визначає базовий лінгвістичний контекст. Для української мови це означає не лише підтримку кириличного алфавіту, а й розробку стратегій обробки код-світчингу (вкраплень латиниці) та суржику. Наступні кроки - фільтрація та очистка тексту - спрямовані на усунення "шуму": видалення HTML-розмітки, URL-адрес, службових символів та іншої нерелевантної інформації, що дозволяє моделі сфокусуватися виключно на значущому контенті. Розділення тексту на

токени (токенізація) є одним із найважливіших кроків, оскільки від обраного методу (наприклад, на рівні слів чи суб-слів) залежить здатність моделі працювати зі складними морфологічними формами та словами, відсутніми у навчальному словнику. Нарешті, формування словника визначає розмірність вхідного простору для моделі та є прямим наслідком токенізації. Помилки або неефективні рішення на будь-якому з цих кроків призводять до втрати важливої семантичної інформації або внесення шуму, що неминуче знижує якість векторних представлень тексту і, як наслідок, обмежує здатність НММ до узагальнення.

Проектування НММ є етапом, на якому закладається її теоретичний потенціал. Правильний вибір архітектури та визначення параметрів нейромережі забезпечують її здатність ефективно вивчати закономірності в даних та розпізнавати емоції. Вибір архітектури (наприклад, згорткова, рекурентна чи трансформерна) має бути обґрунтований специфікою задачі: для аналізу коротких коментарів можуть бути ефективними одні архітектури, тоді як для довгих текстів зі складною наративною структурою - інші. Водночас визначення параметрів, таких як кількість шарів, кількість нейронів, функції активації та механізми регуляризації, безпосередньо впливає на ємність моделі. Недостатня ємність призведе до недонавчання, коли модель нездатна вловити навіть основні залежності в даних. Надлишкова ємність, у свою чергу, створює ризик перенавчання, коли модель "запам'ятовує" навчальні приклади замість того, щоб узагальнювати закономірності, що веде до низької продуктивності на нових даних. Таким чином, етап проектування є процесом пошуку оптимальної складності моделі, яка б відповідала складності задачі та обсягу наявних даних.

Навчання є процесом, під час якого теоретичний потенціал, закладений в архітектуру моделі, реалізується на практиці. Його успішність залежить від кількох ключових факторів. По-перше, це якість бази даних істинних прикладів та стратегія формування навчальної та тестової вибірок, що були детально розглянуті раніше. По-друге, це визначення оптимізаційних

параметрів, таких як алгоритм оптимізації, швидкість навчання та розмір міні-вибірки. Неправильний вибір цих параметрів може призвести до сповільнення або навіть повної зупинки процесу навчання. Нарешті, фінальним кроком є верифікація НММ на відкладеній тестовій вибірці, яка має проводитися за багатокритеріальною системою оцінки, описаною в п. 1.3.

Саме на цьому кроці підтверджується, чи досягла навчена модель необхідного балансу між точністю, обчислювальною ефективністю та надійністю. Отже, ефективність НММ є результатом взаємодії всіх трьох макропроцесів: якісні дані дозволяють правильно спроектованій моделі повною мірою реалізувати свій потенціал під час коректно налаштованого процесу навчання.

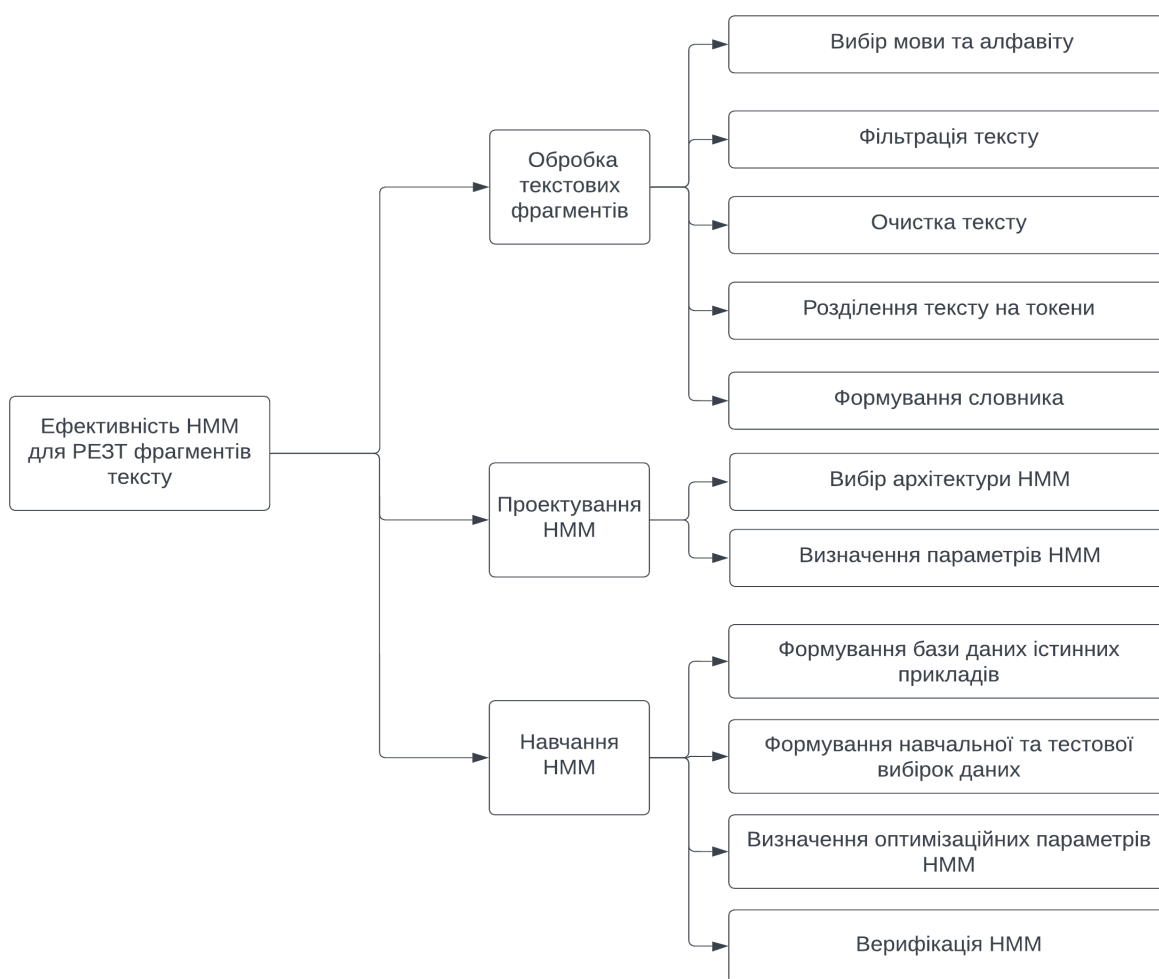


Рис. 2.3. Схема взаємопов'язаних чинників, які визначають ефективність НММ розпізнавання емоційної тональності фрагментів тексту

2.2. Система критеріїв оцінювання ефективності засобів розпізнавання

Практична реалізація запропонованої концептуальної моделі вимагає формалізації процедур верифікації результатів. Згідно із засадами системного підходу, оцінювання ефективності розроблених нейромережових засобів має здійснюватися на багатокритеріальній основі, що охоплює три групи показників які характеризують точність розпізнавання, інтерпретованість прийнятих рішень та обчислювальну ресурсоемність.

Для кількісного оцінювання точності розпізнавання використано стандартний набір метрик, що базуються на елементах матриці похибок (*Confusion Matrix*):

- істинно позитивних (*TP*),
- істинно негативних (*TN*),
- хибно позитивних (*FP*),
- хибно негативних (*FN*).

Базуючись на результатах [55, 56] в якості базових критерії прийнято використовувати:

- *Accuracy* – характеризує загальну частку коректних класифікацій:
- *Precision* – визначає достовірність позитивних прогнозів моделі (частку релевантних об'єктів серед усіх, віднесених до цільового класу)
- *Recall* – характеризує здатність моделі ідентифікувати всі об'єкти цільового класу:
- *F1-score* – гармонійне середнє між точністю та повнотою, що використовується як збалансований критерій оцінки в умовах нерівномірного розподілу класів.

Розрахунок вказаних показників передбачено реалізувати за допомогою виразів (2.3-2.6).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (2.3)$$

$$Precision = \frac{TP}{TP + FP}, \quad (2.4)$$

$$Recall = \frac{TP}{TP + FN}, \quad (2.5)$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \quad (2.6)$$

З метою отримання узагальненої оцінки, що дозволяє проводити ранжування моделей з урахуванням пріоритетів конкретної прикладної задачі, введено інтегральний показник точності класифікації (Q_{total}). Він визначається як зважена лінійна комбінація базових метрик:

$$Q_{total} = \alpha \cdot Accuracy + \beta \cdot Precision + \gamma \cdot Recall + \delta \cdot F1, \quad (2.7)$$

де $\alpha, \beta, \gamma, \delta$ - вагові коефіцієнти ($\alpha + \beta + \gamma + \delta = 1$), значення яких встановлюються залежно від вимог до системи.

Враховуючи вимоги до прозорості функціонування систем штучного інтелекту, систему критеріїв доповнено показниками інтерпретованості. Для верифікації логіки формування рішень нейронною мережею передбачено застосування методів пояснюваного штучного інтелекту (XAI):

1. SHAP (SHapley Additive exPlanations) - для оцінки глобальної важливості ознак (токенів) [57].
2. LIME (Local Interpretable Model-agnostic Explanations) - для локального аналізу окремих прецедентів [58].

Критично важливим етапом є оцінка похибки в процесі навчання, що слугує основою для оптимізації параметрів моделі. Враховуючи, що для кожної окремої емоції задача зводиться до бінарної класифікації, обґрунтованим вибором функції втрат є бінарна перехресна ентропія (англ. Binary Cross-Entropy, BCE). Ця функція має інформаційно-теоретичну інтерпретацію, вимірюючи «відстань» між розподілом істинних міток та розподілом, передбаченим моделлю:

$$L_{BCE}(y, \hat{y}) = -(y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})), \quad (2.8)$$

де y – істинна бінарна мітка (1, якщо емоція присутня; 0, якщо відсутня), $\hat{y}$ – передбачена моделлю ймовірність присутності емоції (число від 0 до 1).

Вираз (2.8) включає дві компоненти. Якщо істинна мітка $y = 1$, другий доданок обнуляється, і втрати залежать лише від $\log(\hat{y})$.

У цьому випадку, якщо модель впевнено передбачає правильну відповідь ($\hat{y} \rightarrow 1$), втрати мінімальні. Якщо ж модель помиляється ($\hat{y} \rightarrow 0$), значення логарифма прямує до $-\infty$, і втрати стають значними, штрафуючи модель. Аналогічно, якщо $y = 0$, працює лише другий доданок, штрафуючи за впевнені хибнопозитивні прогнози.

Оскільки задача розпізнавання емоційної тональності передбачає одночасну класифікацію за M базовими емоціями, і присутність кожної з них вважається незалежною, загальна функція втрат для одного текстового фрагмента обчислюється як сума значень (2.8) по всіх емоціях:

$$L_{total} = \sum_{j=1}^M L_{BCE}(y_j, \hat{y}_j), \quad (2.9)$$

де y_j та $\hat{y}_j$ – істинна мітка та передбачена ймовірність для j -ї емоції відповідно. Мінімізація цього функціоналу дозволяє моделі одночасно навчатися розпізнавати весь спектр базових емоцій.

Оцінювання обчислювальної ресурсоемності, що є критичним фактором для універсальних комп'ютерних систем, базується на апаратно-незалежній метриці FLOPs (*Floating-Point Operations*). Узагальнена оцінка складності моделі визначається через кількість параметрів (P):

$$R \approx k \cdot P, \quad (2.10)$$

де R – умовна ресурсозатратність нейронної мережі (кількість операцій);
 P – загальна кількість параметрів (вагових коефіцієнтів) нейронної мережі;
 k_p – умовний коефіцієнт, що враховує тип нейронної мережі.

Для підвищення точності розрахунку обчислювального навантаження використано підхід, що передбачає врахування щільності зв'язків в шарі нейронів. Для повнозв'язних шарів кількість операцій становить:

$$FLOPs_{fc} \approx 2 \cdot I \cdot O, \quad (2.11)$$

де I – розмірність вхідного вектора, O – кількість нейронів (розмірність вихідного вектора).

Для згорткових шарів:

$$FLOPs_{conv} \approx 2 \cdot H_{out} \cdot W_{out} \cdot C_{out} \cdot C_{in} \cdot K_h \cdot K_w, \quad (2.12)$$

де H_{out}, W_{out} – висота та ширина вихідної карти ознак; C_{in} – кількість вхідних каналів; C_{out} – кількість вихідних каналів (кількість фільтрів); K_h, K_w – висота та ширина ядра згортки.

Системне узгодження суперечливих вимог щодо точності та ресурсоемності реалізується через критерій сумарної ефективності (E_{st}), що інтегрує ефективність розробки (E_p) та ефективність експлуатації (E_{nmm}):

$$E_{st} = F(E_p, E_{nmm}). \quad (2.13)$$

Складова E_p відображає витрати на підготовку даних та навчання. Складова E_{nmm} визначається сукупністю показників інтегральної якості (Q_{total}), рівня інтерпретованості та експлуатаційних характеристик.

Узагальнену схему формування критерію сумарної ефективності наведено на рис. 2.4.

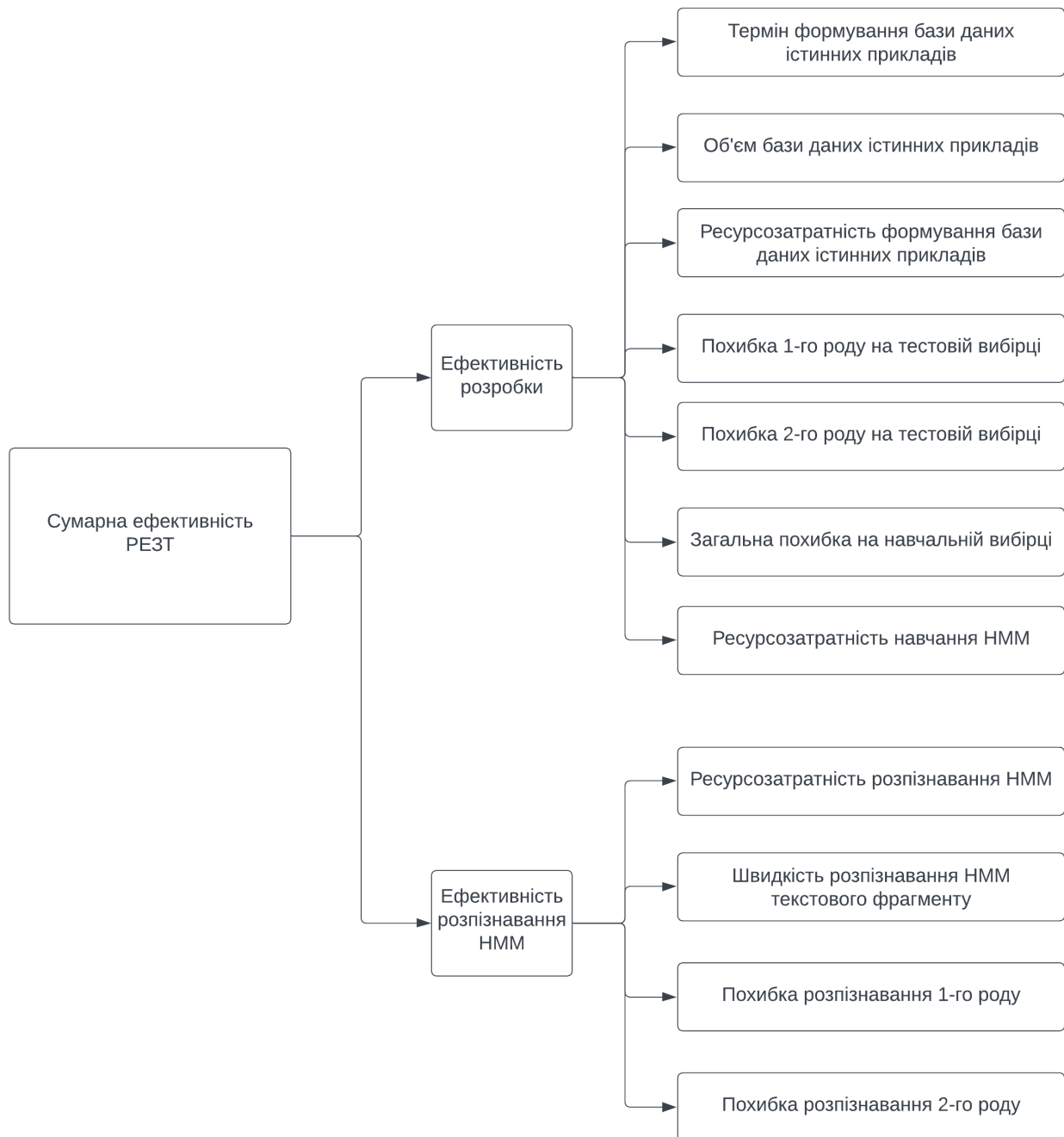


Рис. 2.4. Узагальнена схема формування сумарної ефективності засобів розпізнавання

Розроблена система критеріїв дозволяє формалізувати задачу вибору нейромережевої архітектури як задачу пошуку оптимального співвідношення між затратами на розробку та ефективністю експлуатації відповідно до обмежень цільової платформи.

2.3. Модель побудови нейромережових засобів розпізнавання емоційного забарвлення фрагментів текстів

Розпізнавання емоцій у текстах є багатоаспектною задачею, що поєднує лінгвістичні, когнітивні та культурні чинники. Ефективність систем зростає за рахунок комбінування класичних методів опрацювання мови (лексиконно-правильний аналіз, синтаксичне та прагматичне моделювання) із сучасними глибинними підходами. Гібридні архітектури, які інтегрують трансформери з рекурентними компонентами, демонструють стійкість до полісемії та моделювання довгих залежностей у тексті [59]. Огляди сучасних рішень з аналізу тональності та емоцій узагальнюють переваги багаторівневих представлень, механізмів уваги та процедур адаптації до домену [60]. На рівні інженерії систем доцільною є модульність, що дозволяє варіювати конфігурації компонентів і підлаштовувати модель під специфіку корпусу та задачі [61].

Для україномовних даних проблему ускладнюють багата морфологія (словозміна, словотвір), вільний порядок слів і часте кодування емоцій імпліцитними, культурно зумовленими маркерами. Додаткові виклики формують код-міксинг з російською та англійською, сленг, регіоналізми, а також швидкі зсуви мовної норми під впливом суспільно-політичних подій. За цих умов потрібні регулярне оновлення параметрів і словників, механізми безперервного навчання та контроль дрейфу даних, щоб підтримувати валідність розпізнавання у часі. Обґрунтованою стратегією є перенавчання та доменна адаптація попередньо натренованих (зокрема мультимовних) трансформерів у поєднанні з легковагими спеціалізованими модулями [59, 60].

Концепція ієрархічного розпізнавання та розподілу ролей між модулями, напрацьована в обробці мовлення, може слугувати методологічною опорою під час проєктування та аналізу помилок такої системи [62]. Сукупно це підкреслює необхідність цілісної, гнучкої та оновлюваної моделі процесу розпізнавання емоцій для україномовних текстів, здатної інтегрувати різні

типи нейронних мереж і забезпечувати відтворюваність результатів у часі [59-61].

На основі проведеного аналізу, що підкреслює переваги гібридних та багатозадачних підходів, а також на дослідженнях [59-61], що демонструють ефективність специфічних компонентів для обробки емоцій, у даній роботі пропонується відійти від розробки монолітних архітектур на користь модульного, компонент-орієнтованого підходу.

Така архітектура процесу розпізнавання дозволяє гнучко адаптувати засоби РЕЗТ до унікальних вимог кожного конкретного завдання, а також до обмежень цільової обчислювальної платформи. Замість єдиного, незмінного конвеєра, модель розглядається як система з кількох ключових компонентів, які можна незалежно конфігурувати, оптимізувати або замінювати. Цей підхід передбачає декомпозицію процесу на три основні логічні блоки, які можна налаштовувати відповідно до потреб реалізації, і включає три основні блоки, зображені на рис. 2.5.

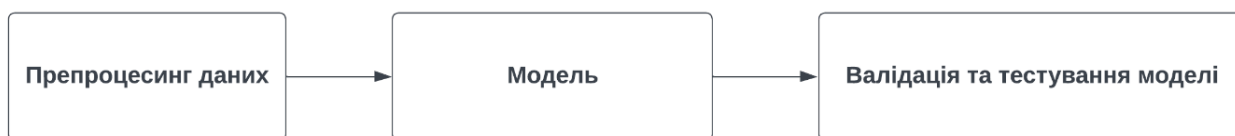


Рис. 2.5. Основні складові моделі процесу розпізнавання емоцій

Основною метою запропонованої моделі побудови нейромережових засобів є опис категоріальної класифікації та кількісної оцінки емоційного забарвлення текстових фрагментів, в рамках одного модуля. Такий підхід є значно більш інформативним, оскільки людські емоції рідко бувають дискретними та абсолютними. Текст може одночасно містити ознаки кількох емоцій різного ступеня інтенсивності. Тому завданням моделі є не просто вибір однієї "домінуючої" емоції, а створення багатовимірного емоційного профілю тексту. Це дозволяє вловлювати значно тонші нюанси та складні емоційні стани, такі як гірко-солодка радість (суміш радості та смутку) або тривожне

очікування (суміш страху та подиву), що є недосяжним для традиційних класифікаторів.

Формально, модель призначена для прогнозування інтенсивності для попередньо визначеного набору з семи базових емоцій, що є комбінацією класичних моделей та практичних потреб аналізу онлайн-комунікації: Радість, Любов, Смуток, Гнів, Страх, Відраза та Нейтральність. Для вхідного текстового фрагмента x модель генерує вихідний вектор y :

$$y = (y_e)_{e \in \mathcal{E}}, \quad (2.14)$$

де $\mathcal{E} = \{\text{Радість, Любов, Смуток, Гнів, Страх, Відраза, Нейтральність}\}$ - цільова множина (таксономія) емоцій.

Кожна компонента y_e цього вектора, що відповідає емоції $e \in \mathcal{E}$, набуває значень у неперервному діапазоні.

Ці значення y_e інтерпретується як передбачена моделлю інтенсивність або ступінь присутності відповідної емоції у тексті x . Значення, близьке до 1, вказує на сильну вираженість емоції, тоді як значення, близьке до 0, свідчить про її відсутність або дуже слабку присутність. Такий регресійний підхід до класифікації дозволяє уникнути жорстких бінарних рішень і надає значно багатшу інформацію для подальшого аналізу.

Важливо підкреслити, що хоча поточна конфігурація орієнтована на ці сім емоцій, архітектура є модульною та розширюваною. Це є однією з ключових переваг запропонованого підходу. Цільова множина емоцій $\mathcal{E}$ не є жорстко закодованою в архітектуру. Завдяки компонентній структурі, для адаптації системи до іншої таксономії (наприклад, більш деталізованої, що включає "провину", "сором", "заздрість") достатньо модифікувати лише вихідний шар нейронної мережі та провести її донавчання на даних з новою розміткою. При цьому модулі попередньої обробки та видобуття ознак можуть залишитися незмінними, що значно спрощує та прискорює процес адаптації моделі до нових, специфічних завдань.

Препроцесинг даних є першим і фундаментальним етапом у запропонованій модульній архітектурі. Цей етап не є простою технічною процедурою; він являє собою комплексний процес перетворення сирих, неструктурованих текстових даних у якісний, стандартизований та машиночитаний формат. Якість виконання цього етапу безпосередньо обмежує потенціал будь-якої, навіть найскладнішої, нейромережевої моделі, що слідує за ним, відповідно до принципу "сміття на вході - сміття на виході".

Процес складається з послідовних стадій обробки, зображених на рис. 2.6, кожна з яких вирішує специфічні завдання, спрямовані на підвищення співвідношення "сигнал-шум" та забезпечення репрезентативності фінального набору даних.

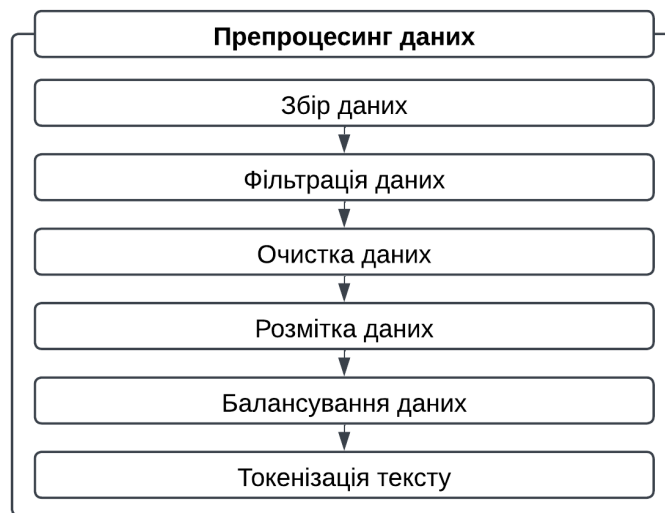


Рис. 2.6. Завдання препроцесингу даних

Завдання 1. Збір даних. Виконання завдання передбачає акумуляцію текстового контенту з гетерогенних джерел. Для адекватного моделювання емоційного ландшафту українського інтернет-простору, джерела мають охоплювати різні стилістичні та жанрові реєстри: від коротких, емоційно насичених дописів у соціальних мережах (Facebook, Telegram, X – Twitter, Instagram) та коментарів під новинними статтями, до більш структурованих текстів, таких як відгуки на товари чи послуги та публіцистичні статті. Особлива увага приділяється диверсифікації вхідних даних для забезпечення

максимальної репрезентативності вибірки. Ця вимога є критично важливою для запобігання доменному перенавчанню (англ. domain overfitting), коли модель добре працює на даних, подібних до навчальних, але різко втрачає ефективність на текстах з інших доменів [63]. Різноманітність контексту, тем, лексики та стилів написання сприяє формуванню більш робастної моделі, здатної до ефективною генералізації.

Завдання 2. Фільтрація даних. Завдання виконується після збору сирого масиву даних настає етап фільтрації, що включає багаторівневий процес селекції релевантного контенту. Його мета - видалити дані, які є непридатними для аналізу або можуть негативно вплинути на процес навчання. На цьому етапі здійснюється відсіювання неукраїнськомовних текстів за допомогою спеціалізованих бібліотек для ідентифікації мови. Далі, видаляються фрагменти недостатньої довжини (наприклад, менше 3-5 слів), оскільки вони зазвичай не містять достатньо контексту для надійної ідентифікації емоції. Останнім кроком є усунення повних або майже повних дублікатів, що запобігає штучному завищенню представленості певних фраз у навчальній вибірці та забезпечує чистоту тестового набору даних. Такий підхід забезпечує формування якісного та унікального корпусу для подальшого аналізу.

Завдання 3. Очистка даних - фокусується на нормалізації та стандартизації безпосередньо текстового контенту. Виконання цього завдання є критичним для забезпечення консистентності вхідних даних та зменшення варіативності, не пов'язаної зі змістом. При цьому передбачається:

- Видалення технічної інформації: усунення HTML-тегів, URL-посилань, імен користувачів (@username), хештегів та інших мета-даних, які не несуть прямого емоційного навантаження.

- Нормалізація пунктуації та символів: приведення тексту до нижнього регістру, видалення зайвих пробілів, уніфікація лапок, а також обробка повторюваних знаків пунктуації (напр., !!! -> !), які можуть бути маркерами інтенсивності.

- Корекція орфографічних помилок та обробка сленгу: за можливості, виправлення очевидних друкарських помилок та розкриття поширених скорочень і сленгових виразів, що дозволяє уніфікувати лексичний склад корпусу.

- Стандартизація кодування символів до єдиного формату (напр., UTF-8) для уникнення технічних проблем на наступних етапах.

Завдання 4. Розмітка (анотування) даних. Завдання полягає у є присвоєнні кожному текстовому фрагменту емоційної мітки (або вектору інтенсивностей). Цей процес традиційно здійснюється шляхом експертної оцінки, де кваліфіковані анотатори, керуючись розробленими інструкціями, визначають емоційне забарвлення тексту. Для оптимізації та масштабування цього процесу в даній роботі передбачається застосування гібридного підходу "людина-в-контурі" (human-in-the-loop) з використанням сучасних великих мовних моделей (LLM). LLM використовуються для генерації початкових, "кандидатських" анотацій, які потім верифікуються, коригуються або відхиляються експертом. Як показують дослідження [53, 64], такий підхід дозволяє значно підвищити ефективність розмітки (на 40-60%) порівняно з повністю ручним процесом, знижуючи часові та фінансові витрати. Виконання завдання 4 є достатньо ресурсозатратним, а якість отриманого результату безпосередньо впливає на якість розпізнавання в цілому.

Завдання 5. Балансування даних. Доцільність виконання даного завдання пояснюється тим, що природний розподіл емоцій у текстах є вкрай нерівномірним (наприклад, тексти з емоцією "радість" або "нейтральні" зустрічаються значно частіше, ніж з емоцією "відраза"). Навчання на таких незбалансованих даних призводить до того, що модель стає упередженою на користь домінуючих класів, ігноруючи рідкісні, але важливі емоції. Для вирішення цієї проблеми застосовуються техніки:

- Недодискретизація (Undersampling): Видалення частини прикладів з мажоритарних класів.

- Передискретизація (Oversampling): Дублювання прикладів з міноритарних класів.

- Генерація синтетичних даних: Створення нових, штучних прикладів для міноритарних класів за допомогою таких методів, як SMOTE [65] або технік текстової аугментації (напр., зворотній переклад).

Цей крок забезпечує оптимальний розподіл класів у навчальній вибірці, що є критичним для запобігання упередженості моделі.

Завдання 6. Токенізація тексту. Токенізація тексту це фінальна частина препроцесингу, що полягає у розділенні суцільного тексту на менші одиниці - токени, які слугуватимуть входом для нейромережевої моделі. Вибір методу токенизації суттєво впливає на якість подальшої векторизації. Існують три основні підходи:

- Токенізація на рівні слів: простий підхід, що страждає від проблеми великого словника та нездатності обробляти слова, відсутні в ньому (out-of-vocabulary, OOV), що є особливо актуальним для морфологічно багатих мов, як українська.

- Посимвольна токенизація: вирішує проблему OOV, але призводить до дуже довгих послідовностей та ускладнює вивчення семантики слів.

- Підслівна токенизація (Subword Tokenization): є сучасним стандартом, що використовується у трансформерних архітектурах. Алгоритми, такі як BPE [66] або SentencePiece [67], розбивають рідкісні слова на менші, значущі частини, зберігаючи при цьому поширені слова цілими. Це дозволяє ефективно працювати зі складною морфологією, неологізмами та друкарськими помилками, знаходячи оптимальний баланс між розміром словника та довжиною послідовностей.

У сукупності виконання вказаних шести завдань формують комплексний конвеєр препроцесингу, який перетворює потік сирих текстових даних на структурований та збалансований набір даних. Саме цей набір даних, що є результатом препроцесингу слугує фундаментом для подальшого проектування, навчання та валідації ефективної НММ РЕТ.

У свою чергу модель ефективних НМЗ розпізнавання містить три основні блоки (див рис 2.7):

1. Визначення набору типових модулів.
2. Визначення значень архітектурних параметрів.
3. Реалізація тренування.

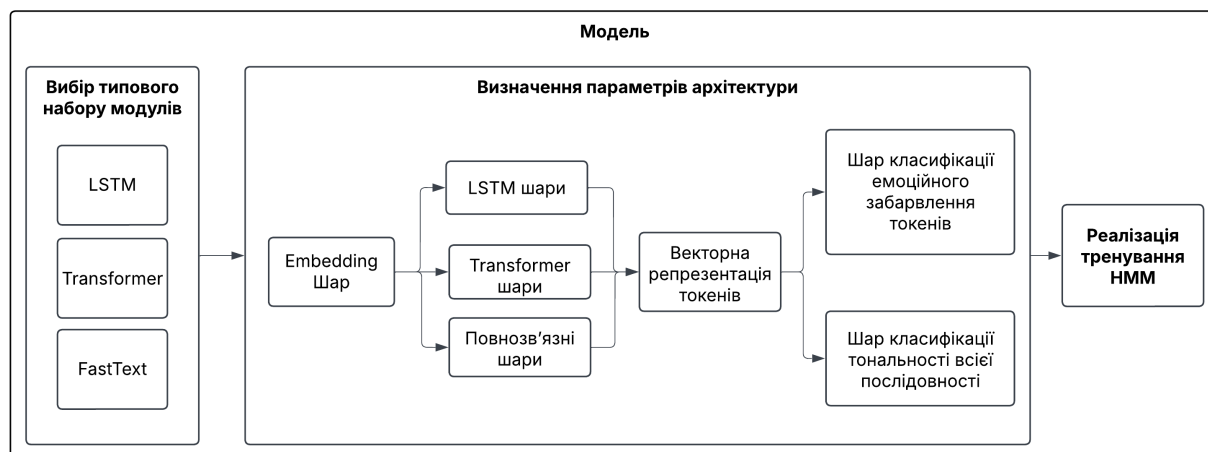


Рис. 2.7. Структурна схема основних процесів моделі побудови неймережових засобів розпізнавання емоційного забарвлення

Центральним компонентом запропонованої модульної системи є блок неймережевого моделювання, відповідальний за вивчення складних закономірностей у текстових даних та, власне, розпізнавання емоцій. Вибір архітектури на цьому етапі є ключовим компромісом між передбачуваною точністю, обчислювальною ефективністю та інтерпретованістю моделі. Модель розпізнавання будується на основі однієї з трьох фундаментальних неймережових архітектур або їх ансамблю (комбінації різних архітектур в одній НММ), що представляють собою еволюційний шлях розвитку підходів до обробки природної мови [68].

Архітектура на основі векторних представлень слів (англ. Word-Embedding) є одним із фундаментальних підходів на основі статичних представлень. Цей підхід, що став класичним після роботи [69], базується на ідеї представлення слів у вигляді щільних, низькорозмірних векторів

(ембедингів), які кодують семантичні відношення між ними. Алгоритми, такі як Word2Vec [70] (у варіантах Continuous Bag of Words (CBOW) або Skip-Gram) та GloVe [71], навчаються на великих нерозмічених корпусах для створення статичних векторних представлень.

Кожне слово w зі словника V відображається у вектор $v_w \in R^d$, де d - розмірність простору вкладень (зазвичай 100-300) див рис. 2.8. Для класифікації цілого фрагмента тексту $T = \{w_1, w_2, \dots, w_n\}$, що складається з n слів, найчастіше використовується агреговане представлення, наприклад, усереднення векторів слів (підхід Bag-of-Embeddings):

$$v_T = \frac{1}{n} \sum_{i=1}^n v_{w_i}. \quad (2.15)$$

Отриманий вектор v_T , що репрезентує узагальнену семантику тексту, подається на вхід відносно простого класифікаційного шару $f_{\text{classifier}}$ (наприклад, логістичної регресії або неглибокого багатошарового перцептрона, MLP) для прогнозування фінального вектора емоцій y . Перевага таких моделей полягає в їхній швидкості навчання та низькій обчислювальній вартості під час інференсу, що робить їх придатними для застосунків реального часу з обмеженими ресурсами. Однак, суттєвим недоліком є ігнорування порядку слів та складного синтаксичного контексту, що є критичним для розпізнавання сарказму чи заперечень (наприклад, "не дуже добре" буде інтерпретовано схоже на "дуже добре"). Тим не менш, цей підхід є надзвичайно корисним як початковий базовий рівень (baseline) для швидкої перевірки гіпотез та оцінки складності задачі [72].

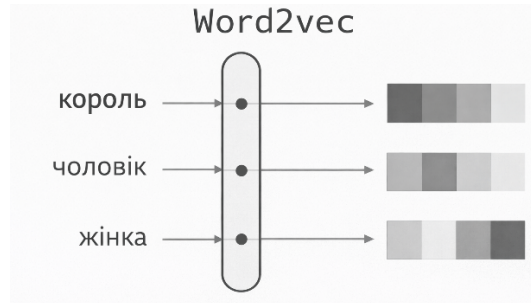


Рис. 2.8. Приклад перетворення слова у чисельний вектор

Рекурентна архітектура з комірками довгої короткочасної пам'яті (LSTM) дозволяє моделювання послідовних залежностей та врахування контексту. Для подолання обмежень контекстної нечутливості були розроблені рекурентні нейронні мережі, зокрема їх значно більш ефективний варіант - мережі з довгою короткочасною пам'яттю (англ. Long Short-Term Memory, LSTM), запропоновані в [73]. LSTM спеціально розроблені для обробки послідовних даних та врахування довготривалих залежностей.

Вхідний текст представляється як послідовність векторів токенів $(x_1, x_2, \dots, x_n)$, де $x_t \in R^d$. LSTM обробляє цю послідовність крок за кроком, оновлюючи на кожному часовому кроці t два вектори стану: прихований стан h_t (короткострокова пам'ять) та стан комірки c_t (довгострокова пам'ять). Цей процес керується системою "вентилів" - нейронних мереж з сигмоїдною активацією, які регулюють потік інформації (див. рис. 2.9) та описується виразами (2.16-2.21). При цьому вираз (2.16) описує вентиль забування, вираз (2.17) описує вхідний вентиль, вираз (2.18) - вихідний вентиль, вираз (2.19) - кандидатний стан, вираз (2.20) - оновлення стану комірки, а вираз (2.21) - оновлення прихованого стану.

$$f_t = \sigma(W_{fx_t} + U * fh * t - 1 + b_f) \quad (2.16)$$

$$i_t = \sigma(W_{ix_t} + U * ih * t - 1 + b_i) \quad (2.17)$$

$$o_t = \sigma(W_{ox_t} + U * oh * t - 1 + b_o) \quad (2.18)$$

$$\tilde{c}_t = \tanh(W_{cx_t} + U * ch * t - 1 + b_c) \quad (2.19)$$

$$c_t = f * t \odot c * t - 1 + i_t \odot \tilde{c}_t \quad (2.20)$$

$$h_t = o_t \odot \tanh(c_t) \quad (2.21)$$

де W, U - матриці ваг, b - вектори зсувів, σ - сигмоїдна функція, а $\odot$ - покомпонентне множення, h_n - фінальний прихований стан.

Зазначимо, що h_n використовується для підсумкової класифікації.

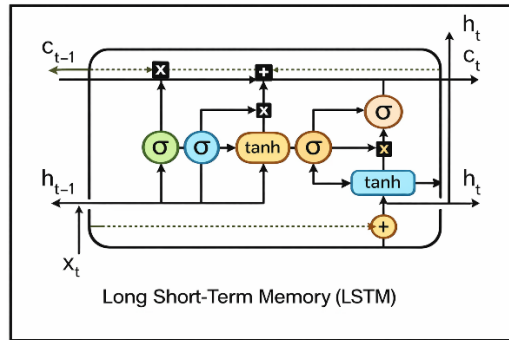


Рис. 2.9. Приклад візуалізації операцій в одній комірці LSTM[73]

Моделі на основі LSTM пропонують значно кращу контекстуальну виразність порівняно з простими моделями, що робить їх придатними для більшості практичних завдань РЕТ, зберігаючи прийнятний баланс між точністю та обчислювальними вимогами.

Архітектура на основі механізму уваги - Трансформер (Transformer) мають переваги перед вище згаданими архітектурами у глибокому контекстуальному розумінні вхідного тексту. Трансформери, представлені у роботі [74], здійснили революцію в обробці природної мови та стали стандартом де-факто для задач, що вимагають глибокого розуміння мови. Їх ключовою інновацією є механізм само-уваги (англ. self-attention), який дозволяє моделі паралельно зважувати важливість кожного токена у вхідній послідовності відносно кожного іншого токена, відмовляючись від рекурентної обробки. Послідовність tokenів $(x_1, \dots, x_n)$, доповнена позиційними кодуваннями $P = (p_1, \dots, p_n)$, що дозволяє враховувати позиція

токена у вхідній послідовності, формує вхідні представлення $X_{input} \in R^{n \times d}$.

Ключове обчислення має вигляд:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (2.22)$$

де Q - запит, K - ключ, V - значення, d_k - розмірність ключа.

При цьому Q , K , V є лінійними проєкціями вхідних представлень X_{input} , а d_k використовується для масштабування. Багатооголова увага (Multi-Head Attention, МНА) дозволяє моделі одночасно зосереджуватися на різних аспектах синтаксичних та семантичних зв'язків. Кожен блок трансформера поєднує МНА, позиційно-залежні нейронні мережі прямого поширення (FFN), нормалізацію за шарами та залишкові з'єднання.

Схема взаємодій компонентів в архітектурі трансформера зображена на рис. 2.10.

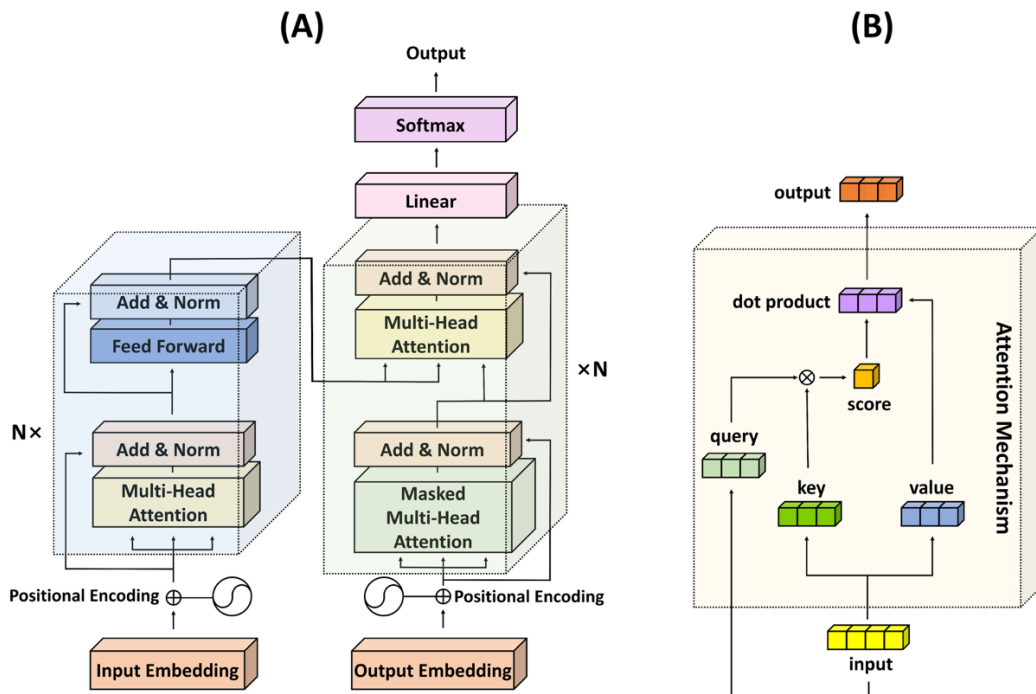


Рис. 2.10. Схема операцій обчислень механізму уваги в архітектурі Transformer

Попередньо навчені на текстових масивах, моделі на основі трансформерів відмінно справляються з розумінням найтонших емоційних нюансів, полісемії та складного контексту, досягаючи найвищих результатів точності. Однак, їхня обчислювальна складність $O(n^2 \cdot d)$ робить їх надзвичайно вимогливими до обчислювальних ресурсів як для навчання, так і для інференсу.

Після вибору базового архітектурного модуля (статичного, рекурентного або трансформерного) наступним кроком є його параметризація - визначення конкретних гіперпараметрів та конфігурації шарів (блок «побудова моделі» на рис. 2.6). Цей етап є критично важливим, оскільки він визначає обчислювальну глибину, ємність та, як наслідок, здатність моделі до навчання та узагальнення. Процес параметризації базується на багаторівневій структурі, що включає диференційовані компоненти обробки тексту, починаючи з фундаментального шару вкладень і закінчуючи спеціалізованими класифікаційними блоками.

Фундаментальним елементом будь-якої архітектури для обробки природної мови є шар вкладень. Його головна функція – перетворення дискретних текстових токенів (слів або підслів) у щільні, неперервні вектори фіксованої розмірності, що дозволяє нейронній мережі працювати з ними як з числовими даними. Цей шар є не просто технічним перетворенням, а першим етапом вилучення семантичних ознак, де близькість векторів у просторі вкладень відображає семантичну схожість відповідних токенів. Конфігурація цього шару адаптується відповідно до обраного базового модуля:

- Word2Vec використовуються статичні, контекстуально-незалежні лексичні векторні представлення, що описуються виразом (2.15). Ці вкладення можуть бути попередньо навчені на великих загальних корпусах (напр., українська Вікіпедія) або навчатися з нуля на цільовому корпусі даних. Їхня головна перевага - обчислювальна простота, а недолік - нездатність розрізняти значення омонімів (напр., слово "коса" матиме однаковий вектор незалежно від контексту).

- LSTM зазвичай застосовуються попередньо навчені статичні вкладення, які далі донавчаються у процесі тренування основної моделі. Це дозволяє початковим семантичним представленням адаптуватися до специфіки емоційної лексики та контексту, присутнього у навчальному наборі даних, що підвищує якість фінальної моделі.

- Трансформер реалізує найбільш складний підхід. Окрім самих вкладень токенів, обов'язково використовуються позиційні вкладення для кодування інформації про порядок слів, що описується виразом (2.22). Найголовніше, що вихідні представлення цього шару не є статичними; вони є контекстуальними, тобто фінальний вектор кожного токена залежить від усього речення, що дозволяє моделі розрізняти полісемію та вловлювати тонкі семантичні нюанси.

- Після отримання початкових векторних представлень необхідно визначити кількість та тип прихованих шарів, що формують "обчислювальне ядро" моделі та відповідають за вивчення ієрархічних ознак.

Рекомендації, що базуються на працях, [73 - 78], є такими:

- Шари довгої короткочасної пам'яті (LSTM) ефективно вловлюють довготривалі залежності у послідовних даних. У задачах класифікації тексту поширеними є конфігурації з одного-трьох шарів. Один шар здатний виявити базові послідовні патерни. Другий шар, отримуючи на вхід послідовність прихованих станів першого, може вивчати більш складні, абстрактні залежності. Часто ці шари реалізуються у двонаправленому (BiLSTM) варіанті для одночасної обробки контексту як у прямому, так і у зворотному напрямках, що є особливо корисним для розуміння заперечень та складних синтаксичних конструкцій. Додавання більшої кількості шарів (понад три) рідко дає значний приріст точності, але суттєво збільшує обчислювальну складність та ризик перенавчання.

- Блоки кодувальника на базі трансформера реалізують (2.17-2.21) для паралельної обробки глобальних контекстуальних зв'язків. Кількість шарів (блоків) є ключовим гіперпараметром, що визначає "глибину" розуміння мови.

У сучасній практиці найчастіше використовуються попередньо навчені моделі, де кількість шарів вже зафіксована: від шести у полегшених архітектурах (наприклад, DistilBERT) до дванадцяти у базових версіях (BERT-base) та двадцяти чотирьох і більше у великих моделях (BERT-large) [74, 75]. Для задач аналізу емоцій стандартною практикою є використання моделей з їхньою початковою кількістю шарів (наприклад, 12 для bert-base-multilingual-cased), оскільки це дозволяє повною мірою використати їхні потужні, попередньо навчені мовні репрезентації.

- Повнозв'язні шари (Fully-connected layers), або багат шаровий перцептрон (MLP), зазвичай розташовуються наприкінці архітектури і виконують функцію класифікаційної голови (classification head). Вони приймають на вхід агрегований векторний образ тексту та виконують фінальні нелінійні перетворення для відображення його у простір цільових класів емоцій. Типова конфігурація включає один або два повнозв'язні шари з функцією активації ReLU перед фінальним вихідним шаром, що є ефективним компромісом між виразністю та ризиком перенавчання.

Фінальним кроком в блоці побудови моделі є вибір механізми класифікації між класифікацією окремих токенів та класифікацією всього тексту. Нехай $T = \{t_1, t_2, \dots, t_n\}$ позначає послідовність вхідних токенів, на які було попередньо розбито вхідний текст. Після обробки ми отримуємо матрицю векторних представлень $V = \{v_1, v_2, \dots, v_n\}$, де $v_i \in R^d$ є d -вимірним контекстуальним вектором токена t_i . На основі цієї матриці реалізуються два паралельні механізми класифікації для забезпечення як глобальної, так і локальної оцінки:

- Класифікатор емоцій на рівні послідовності (E_{seq}). Цей компонент аналізує загальну емоційну динаміку всього фрагмента тексту. Він спершу агрегує всю матрицю V в єдиний контекстний вектор за допомогою функції g (наприклад, використовуючи вихід, що відповідає спеціальному токenu $[CLS]$ у трансформерах, або застосовуючи mean/max pooling), а потім класифікує його за допомогою функції f_{seq} :

$$E_{seq} = f_{seq}(g(V)), \quad (2.23)$$

де $g: R^{n \times d} \rightarrow R^m$ агрегує інформацію, а $f_{seq}: R^m \rightarrow R^k$ відображає її у простір k класів емоцій.

- Класифікатор емоцій на рівні токена (E_{token}). Цей компонент виконує гранулярний аналіз, прогножуючи емоційні характеристики для кожного окремого токена у послідовності. Це дозволяє ідентифікувати, які саме слова є носіями емоційного забарвлення:

$$E_{token} = \{f_{token}(v_1), f_{token}(v_2), \dots, f_{token}(v_n)\}, \quad (2.19)$$

де функція $f_{token}: R^d \rightarrow R^k$ застосовується до кожного вектора токена v_i окремо.

Таким чином, крок параметризації моделі забезпечує як глобальну оцінку емоційної тональності всього тексту (2.18) так і локальну, пояснювальну оцінку для кожного токена (2.19). Такий подвійний підхід не лише підвищує загальну точність та гнучкість системи, але й відкриває можливості для кращої інтерпретації її рішень, що є надзвичайно важливим для побудови надійних та прозорих систем штучного інтелекту.

Останнім завданням, що описує модель побудови нейромережових засобів для розпізнавання емоційного забарвлення та тональності являється валідація навченої НММ, її тестування та інтерпретація. Його мета - не лише констатувати досягнуті показники якості, але й отримати глибоке розуміння поведінки моделі, її сильних та слабких сторін, а також перевірити її відповідність наперед визначеним функціональним та нефункціональним вимогам.

Як показано на схематичному представленні (рис. 2.11), це завдання розділяється на декілька окремих підзавдань, що охоплюють кількісну оцінку, візуальний аналіз та якісну інтерпретацію результатів. Основою етапу є тестування моделі на відкладеній, раніше не баченій тестовій вибірці. Це

єдиний спосіб отримати об'єктивну оцінку здатності моделі до узагальнення. У процесі тестування модель генерує прогнози для кожного текстового фрагмента з тестового набору, які потім порівнюються з еталонною розміткою.

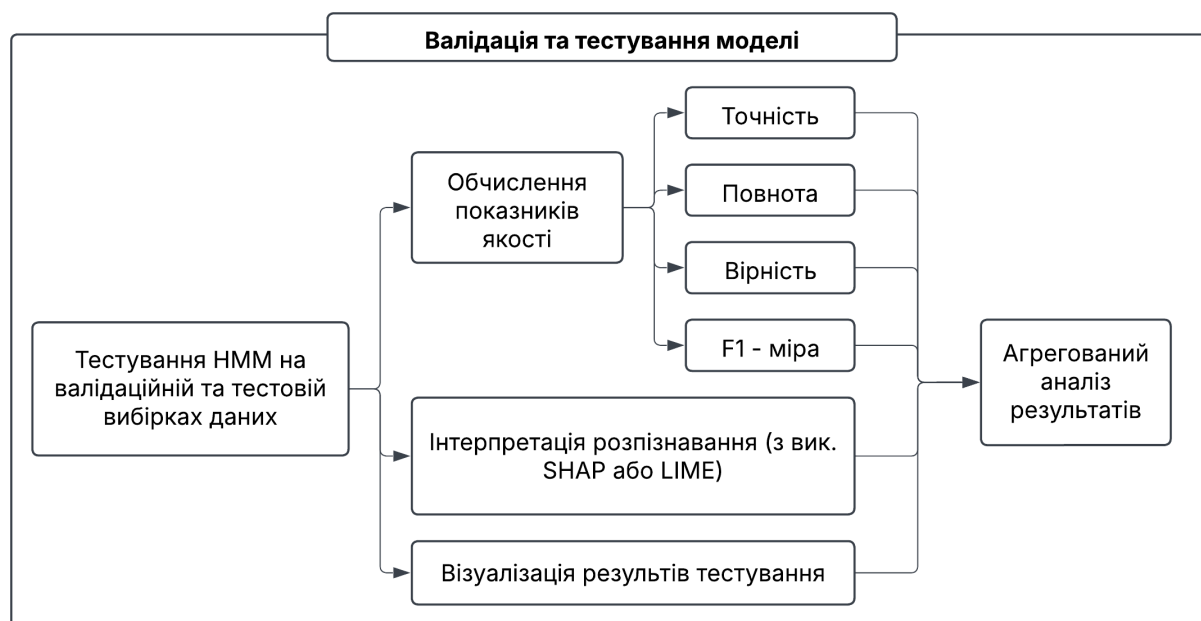


Рис. 2.11. Схема виконання завдання валідації та тестування моделі

Результатом цього порівняння є результат валідації, який агрегується у вигляді матриці плутанини та використовується для розрахунку метрик означених в п. 2.2. Фінальним кроком виконання означеного завдання є аналіз результатів, який синтезує інформацію, отриману з кількісних метрик та методів інтерпретації.

Цілі цього аналізу полягають в наступному:

- Верифікація засвоєних патернів. Перевірка, чи відповідають виявлені за допомогою SHAP та LIME закономірності лінгвістичним та психологічним принципам вираження емоцій, чи модель не покладається на хибні кореляції та артефакти в даних.
- Виявлення упереджень (biases). Аналіз, чи не робить модель систематичних помилок для певних тем, стилів мовлення або соціальних груп, представлених у даних.

- Формування висновків. Надання практично значущих висновків для вдосконалення моделі. Наприклад, якщо аналіз помилок показує, що модель погано розпізнає сарказм, це може стати підставою для збору та анотування додаткових даних, що містять іронічні висловлювання.

- Підвищення прозорості системи. Надання кінцевим користувачам та зацікавленим сторонам зрозумілих пояснень щодо роботи системи, що є ключовим для її відповідального впровадження.

Поєднання надійних метрик продуктивності, сучасних методів інтерпретації та глибокого аналізу результатів забезпечує всебічну валідацію можливостей моделі. Такий комплексний підхід перетворює процес тестування з простої констатації факту на інструмент для ітеративного вдосконалення, що сприяє створенню більш точних, надійних та прозорих систем аналізу емоцій в україномовних текстах.

2.4. Висновки до другого розділу

У другому розділі вирішено наукове завдання дослідження, що полягає у розвитку методологічної бази розпізнавання емоційної тональності текстових фрагментів з використанням нейромережевих засобів в універсальних комп'ютерних системах.

Основні результати розділу наступні:

- Розроблено концептуальну модель розпізнавання емоційної тональності фрагментів тексту, яка декомпозує процес створення системи на взаємопов'язані етапи. За рахунок системного узгодження параметрів нейромережевої архітектури з обмеженнями обчислювальних ресурсів та лінгвістичною специфікою навчальних даних, модель забезпечила формалізований опис напрямків досліджень та стала основою для проектування адаптивних засобів розпізнавання.

- Удосконалено систему критеріїв оцінювання ефективності засобів розпізнавання. На відміну від традиційних підходів, запропонована система

базується на розширеній множині метрик, що включає показники якості (точність, повнота, F1-міра), інтерпретованості (на основі методів SHAP) та ресурсоемності (FLOPs). Це дозволило визначити інтегральний критерій сумарної ефективності, який забезпечує можливість комплексного оцінювання засобів розпізнавання з позицій точності, прозорості прийняття рішень та обчислювальних витрат на етапах розробки й експлуатації.

- Розроблено модель побудови нейромережових засобів розпізнавання емоційної тональності. Модель ґрунтується на модульній архітектурі, що інтегрує процедури токенізації на основі підслівних одиниць (для вирішення проблеми морфологічної варіативності української мови) та формалізовану множину конструктивних параметрів (розмірність вкладень, кількість шарів, тип механізму уваги). Такий підхід забезпечує адаптацію нейромережових засобів до лінгвістичної специфіки текстів та ресурсних обмежень конкретних умов застосування, дозволяючи створювати масштабовані рішення.

Сукупність розроблених моделей і системи критеріїв дозволила доповнити методологічну базу, яка є теоретичним фундаментом для розробки методу розпізнавання та проведення експериментальних досліджень у наступних розділах дисертаційної роботи.

РОЗДІЛ 3

МЕТОД РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ

3.1 Процедура формування вхідних і вихідних даних

Правильна підготовка вхідних даних є ключовою для успішного розпізнавання емоційної тональності тексту. Як зазначалося в розділі 2, вибір способу представлення тексту значною мірою визначає, які семантичні й синтаксичні ознаки можуть засвоїти нейромережеві моделі. У цьому підрозділі розглянемо кілька підходів до векторизації текстових даних – від класичних Bag-of-Words [79] методів (таких як TF-IDF) до статичних ембеддингів (Word2Vec, GloVe) і сучасних контекстуальних представлень (типу BERT). Окрему увагу приділимо токенизації (включно з підслівною) та перетворенню тексту у числові індекси словника, а також пояснимо структуру словника і взаємозв'язок із embedding-матрицею [80]. Це забезпечить концептуальне підґрунтя для підготовки даних відповідно до методології, викладеної у матеріалах другого розділу.

Найпростішим способом представлення тексту для моделі машинного навчання є підрахунок частот слів – підхід Bag-of-Words (BoW). У BoW-технології кожен документ (текстовий фрагмент) представлено великим розрідженим вектором розміру $|V|$, де $|V|$ – розмір словника корпусу: позиція відповідає певному слову, а значення – частоті або наявності цього слова в документі. Наприклад, у двох реченнях «Цей фільм дуже хороший» і «Цей фільм не хороший» BoW-вектори відобразять лише частотну присутність слів «цей, фільм, дуже, хороший, не» тощо, не враховуючи порядку слів. Для покращення BoW-представлення часто використовують зважування TF-IDF, яке збільшує вагу слів, характерних для конкретного документа, і зменшує вагу занадто загальних слів. Формально TF-IDF вагу слова w в документі d можна подати, як добуток його частоти в цьому документі (Term Frequency, TF) на

обернену частоту документів з цим словом (Inverted Document Frequency, IDF). Це знижує вплив стоп-слів на кшталт «це», «і», «в» та акцентує значущі терміни [77, 78].

Водночас класичні Bag-of-Words моделі мають суттєві обмеження. По-перше, вектори є надзвичайно розмірними (розрядність дорівнює розміру словника, який для великого корпусу може сягати сотні тисяч слів). Це призводить до розрідженості представлень і потребує більше даних для надійного навчання моделей. По-друге, BoW підхід повністю ігнорує порядок слів та контекст: він враховує лише факт наявності слова, але не його значення у реченні. Через це складні мовні звороти можуть лишитися нерозпізнаними. Класичний приклад: речення «How could anyone sit through this movie?» не містить явно «негативних» слів, проте загалом передає негативне враження [80].

Для людини це очевидно, але модель BoW не зможе вловити тонкий сарказм чи риторичне запитання – їй бракує семантичного розуміння. Таким чином, незважаючи на успіхи BoW у базових завданнях, для розпізнавання емоційної тональності складніших висловлювань вони часто недостатні.

Більш інформативне представлення тексту забезпечують семантичні ембеддинги – щільні вектори для слів, навчені на великих корпусах мовлення. Як обговорено в розд. 2, алгоритми на кшталт Word2Vec (варіанти CBOW та Skip-Gram) та GloVe дозволяють зіставити кожному слову w зі словника V сталий вектор $v_w \in R^d$, де d – розмірність простору ознак. Такі моделі навчаються на базі статистики співзустрічальності слів: слова, що часто трапляються в схожих контекстах, отримують близькі векторні координати у просторі. Це означає, що ембеддинги здатні кодувати семантичні взаємозв'язки. Відомим прикладом є аналогії типу «король» – «чоловік» + «жінка» $\approx$ «королева», які можна отримати через арифметичні операції над відповідними векторами. Таким чином, на відміну від розріджених TF-IDF, слова представлені як щільні низьковимірні вектори, що зберігають значеннєву

близькість: синоніми чи тематично пов'язані слова будуть розташовані поруч у цьому векторному просторі.

Оскільки статичні ембеддинги призначені для окремих слів, постає питання формування векторів цілих речень чи документів. Найпростіша стратегія – усереднення векторів слів або інший тип масштабування. В базовому варіанті запропоновано використовувати підхід до усереднення векторів слів, що описується за допомогою виразу (2.15). Такий підхід згортає змінну довжину тексту в фіксований вектор сталого розміру d , який вже можна подавати на вхід класифікатору. Наприклад, для речення «Не дуже добре» отримаємо середній вектор з ембеддингів слів «не», «дуже», «добре». Цей вектор узагальнює семантику фрази і може бути поданий до простого нейронного класифікатора або навіть логістичної регресії для передбачення емоційного класу або рівня. Завдяки низькій розмірності d (сотні замість десятків тисяч) такі представлення значно компактніші, ніж BoW.

Статичні векторні представлення поєднують у собі простоту реалізації та певну «обізнаність» про значення слів. Їх можна попередньо навчити на великих не розмічених текстах (наприклад, українська Вікіпедія чи веб-корпус) і потім застосувати до конкретної задачі розпізнавання емоцій. Головна перевага – це обчислювальна ефективність: отримання векторів через таблицю ембеддингів відбувається дуже швидко, а навчання простих моделей на їх основі потребує менше ресурсів. Як зауважено в другому розділі, такі рішення придатні для реального часу та обмежених апаратних умов. Проте статичні ембеддинги мають принциповий недолік: вони не враховують контекст використання слова. Один і той самий вектор призначений для слова незалежно від того, в якому значенні або оточенні воно вжито. Це особливо проблематично для багатозначних слів. Наприклад, українське слово «коса» (яке може значити і зачіску, і сільськогосподарське знаряддя, і піщану косу) матиме один-єдиний вектор у Word2Vec-словнику, що не дозволяє розрізнити смисли в конкретному реченні. В задачах аналізу емоцій така контекстна нечутливість може приводити до помилок. Крім того, при побудові середнього

вектора фрази втрачається порядок слів: таким методом важко врахувати заперечення чи інтенсивність. Як результат, модель може сплутати «дуже добре» і «не дуже добре», адже середні вектори цих фраз будуть близькими (ті ж самі компоненти, просто у другій додано вектор заперечного «не»). Емпірично підтверджено, що ігнорування порядку та синтаксису є критичним для тональних завдань – наприклад, сарказм зазвичай неможливо виявити без контекстної інформації. Підсумовуючи, статичні ембеддинги значно перевершують BoW-бейзлайн у здатності до семантичного узагальнення (модель може «зрозуміти», що “сумний” і “засмучений” близькі за емоційним змістом). Водночас вони поступаються сучаснішим методам, які навчають представлення слова в залежності від його оточення.

Новітній етап у розвитку методів обробки тексту – це появу контекстуальних ембеддингів, які динамічно змінюються залежно від повного речення. BERT та подібні моделі є представниками навчаються на завданні мовного моделювання: прогнозують пропущені слова або наступні речення, опрацьовуючи великі обсяги тексту. Завдяки цьому вони набувають узагальнене розуміння мови, включно з нюансами емоційного забарвлення. Ключова відмінність від статичних підходів полягає в тому, що кожне слово отримує різні векторні представлення залежно від контексту. Наприклад, слово «коса» у різних реченнях матиме різні вектори, що дозволить моделі правильно інтерпретувати емоцію (скажімо, «коса дівчини викликає радість» проти «коса, як зброя викликає страх» – BERT розрізнить ці ситуації) [76].

Контекстуальні моделі потребують спеціальної підготовки тексту. Перед поданням у трансформер речення токенизується на підслова (детально про підслівну токенизацію – далі), додаються службові токени (напр. *[CLS]* на початок для позначення початку послідовності та *[SEP]* в кінець), а також генеруються позиційні індекси для врахування порядку слів. На першому шарі модель має звичайну embedding-матрицю для перетворення tokenів у початкові векторні ознаки розміру d (для BERT_base $d=768$). Але далі ці вектори багаторазово трансформуються в глибоких шарах самоуваги,

обмінюючись інформацією між всіма позиціями. На виході отримуємо високорівневі ознаки: наприклад, фінальний *hidden-state* відповідного до [CLS] токена часто використовується як узагальнений вектор усього речення, що містить інтегровану інформацію про тональність і емоції. Застосування такої схеми дає змогу моделі врахувати складні мовні залежності – від заперечень і сарказму до стилістичних нюансів авторського мовлення.

Водночас, плата за таку високу якість – це значно більші обчислювальні витрати. Моделі на кшталт BERT містять сотні мільйонів параметрів, тому їх використання вимагає більше пам'яті та часу. Для порівняння, FastText-ембеддинги (300-вимірні статичні) працюють швидше, ніж BERT-ембеддинги (768-вимірні контекстуальні), особливо при обробці великих корпусів або в режимі реального часу.

У практичних сценаріях це породжує компроміс: якість vs швидкодія. Деякі дослідники навіть пропонують гібридні рішення – наприклад, використання тільки першого шару BERT як фіксованих ембеддингів (що вже дає певний виграш над Word2Vec) або стискання вимірності BERT-векторів методом. Перед тим як текст потрапить до описаних моделей, його потрібно перетворити з рядка символів у послідовність чисел. Цей процес називається токенізацією та індексацією. Токенізація розбиває сирий текст на окремі токени – елементи, що надалі оброблятимуться моделлю. В залежності від підходу токенами можуть бути слова, частини слів або навіть окремі символи. Правила токенізації мають значення особливо для аглютинативних і флективних мов (таких як українська), де слова можуть набувати багатьох форм. Найбільш інтуїтивно простий підхід – токенізація на рівні слів. Вона полягає у розділенні тексту за пробілами та розділовими знаками, виділяючи цілі слова як окремі токени. Такий токенізатор перетворить речення «Фільм був надзвичайно цікавим!» на список токенів: ["Фільм", "був", "надзвичайно", "цікавим", "!", ""]. Словесна токенізація є зручною та легко інтерпретованою, але для морфологічно багатих мов, як-от українська, вона породжує дві суттєві проблеми.

По-перше, це призводить до створення надзвичайно великого словника. Через те, що слова змінюються за відмінками, числами, родами тощо, простий словник може налічувати сотні тисяч унікальних словоформ.

По-друге, що є прямим наслідком першої проблеми, це створює проблему невідомих слів (англ. Out-of-Vocabulary, OOV). Слова, що не траплялися у навчальному корпусі (наприклад, неологізми, рідкісні терміни або просто інші граматичні форми), модель не зможе обробити. Формально, процес перетворення послідовності tokenів $(w_1, w_2, \dots, w_n)$ на послідовність числових індексів $(i_1, i_2, \dots, i_n)$ для подачі в нейронну мережу описується функцією відображення, що враховує наявність слова у фіксованому словнику V :

$$i_j = \begin{cases} \text{index}(w_j), & \text{якщо } w_j \in V \\ \text{index}(\langle UNK \rangle), & \text{якщо } w_j \notin V \end{cases} \quad (3.1)$$

де $\text{index}(w_j)$ - унікальний індекс слова w_j у словнику, а $\langle UNK \rangle$ - спеціальний token для всіх невідомих слів. Коли модель зустрічає token $\langle UNK \rangle$ вона втрачає всю семантичну інформацію про оригінальне слово, що неминуче знижує якість розпізнавання емоцій, особливо якщо емоційне навантаження несло саме невідоме слово.

Сучасні системи віддають перевагу гібридному підходу – токенизації на рівні підсловних морфем або навіть окремих символів. Ідея полягає в тому, щоб мати відносно невеликий фіксований словник tokenів (наприклад, 30 тисяч), з якого можна скласти будь-яке слово. Найпопулярніший метод – Byte-Pair Encoding (BPE), запропонований у [66]. BPE починає зі списку всіх окремих символів (літер) як початкових tokenів і поступово злипає найчастіші пари символів у нові токени. Після десятків тисяч ітерацій формується оптимальний набір підсловних фрагментів, які часто зустрічаються. Завдяки цьому алгоритму досягається компроміс: словник має фіксований заданий розмір, але будь-яке рідкісне чи складене слово може бути розбите на знайомі

підслова. Наприклад, слово «надзвичайно» може бути розділене на підслова «над», «звичайно», а вигадане слово «гіперцікаво» – на токени «гіпер», «цікаво», які, ймовірно, вже є в словнику. Таким чином, модель ніколи не зіткнеться з повністю невідомим набором символів – відкритий словник досягається через композицію підслівних юнітів. BPE і подібні алгоритми (WordPiece[81] у BERT, Unigram[82] у SentencePiece) суттєво покращують роботу з мовами складної морфології та зменшують проблему OOV (out-of-vocabulary). FastText, згаданий вище, теж використовує інформацію про символні n -грами всередині слів, що дозволяє йому навчати вектори навіть для нечастотних або хибно написаних слів. Векторизація tokenів та embedding-матриця. Незалежно від рівня токенизації (слова чи підслова), наступним кроком є перетворення кожного токена у числовий ідентифікатор. Формується словник tokenів – впорядкований перелік усіх можливих tokenів, наприклад, у вигляді відображення: $\text{token} \rightarrow \text{індекс}$. Типовий приклад: $\{ "<PAD>":0, "<UNK>":1, "<CLS>":2, "<SEP>":3, \text{"не"}:4, \text{"дуже"}:5, \text{"добре"}:6, \dots \}$. Спеціальні службові токени (PAD для заповнення, UNK для невідомих, тощо) теж мають зарезервовані індекси. Після цього кожен токен у тексті замінюється своїм індексом.

Шар вбудовувань – це проста матриця розміру $|V| \times d$ (кількість tokenів на розмір ембеддинга). Подаючи індекс i , ми фактично вибираємо i -й рядок цієї матриці, тобто вектор розмірності d , асоційований з токеном

Якщо ембеддинги попередньо навчені (наприклад, завантажені Word2Vec), то значення матриці ініціалізуються відповідними числами і можуть лишатися фіксованими або тонко донавчатися під час тренування моделі. У випадку end-to-end навчання з нуля, елементи матриці ініціалізують випадково і оновлюють градієнтним спуском разом з іншими вагами моделі.

Отримані вектори (рядки таблиці) утворюють тензор розміру $n \times d$, де n – кількість tokenів у реченні. Такі тензори можуть доповнюватися під час батчування до фіксованої довжини послідовності (для цього використовується [PAD] токен «пустий» токен). Векторні представлення разом із позиційним

кодуванням (у трансформерах) подаються на вхід моделі, яка вже безпосередньо навчається розпізнавати емоційне забарвлення.

Формат вихідних даних (цільових міток) визначається постановкою задачі розпізнавання емоцій. У нашому дослідженні (див. розд. 2) розглядаються дві взаємопов'язані, але різні за природою задачі: визначення тональності (сентимент-аналіз) і визначення багатовимірного емоційного профілю тексту. Відповідно, використовуються два типи вихідних міток: категоріальні, що відносять фрагмент до одного з класів тональності (позитивної, негативної тощо), та багатовимірні, що описують інтенсивність кількох базових емоцій одночасно. Кожен тип міток має свої особливості формування, підходи до кодування і відповідні функції втрат при навчанні моделі. Розглянемо їх окремо.

Категоріальні мітки – це дискретні класи, що характеризують загальну тональність тексту. Найчастіше використовується три класи: позитивна, негативна або нейтральна тональність. Іноді додаються проміжні або більш деталізовані категорії (наприклад, «дуже позитивна», «дуже негативна» для градуальної шкали або спеціальні класи на кшталт «сарказм»). Для прикладу, речення «Мені сподобався цей фільм» отримає мітку позитивна, а «Фільм розчарував, це провал» – негативна. У корпусі навчальних даних такі мітки можуть бути представлені в текстовому вигляді або ж закодовані чисельно (наприклад, 2=позитивна, 1=нейтральна, 0=негативна) – головне, що кожен фрагмент має тільки один тональний клас. Це класична постановка завдання класифікації, де кожен приклад належить до одного з K взаємовиключних класів.

One-hot кодування і цільові вектори [83]. При подачі в нейромережу категоріальні мітки часто перетворюють у формат one-hot вектора довжини K . Для трьох класів це буде вектор з трьох компонент, де позиція, що відповідає правильному класу, позначається 1, інші – 0. Наприклад, мітка нейтральна можна закодувати як $(0; 1; 0)$, якщо домовитися про порядок [негативна, нейтральна, позитивна]. Таке кодування зручне для використання на виході

мережі з K нейронів: під час навчання мережа генерує оцінки належності до кожного класу (як правило, застосовується softmax-активація для перетворення сирих логітів у ймовірності, що сумуються до 1). Цільовий one-hot вектор порівнюється з виходом моделі.

Для багатокласової тональної класифікації стандартною функцією втрат є крос-ентропія між розподілом, що спрогнозувала модель, та «вірогідним» розподілом, що відповідає правильному класу. Крос-ентропійна втрата штрафувє модель за відхилення ймовірності правильного класу від 1.0. Мінімізація цієї втрати еквівалентна максимізації правдоподібності правильних класів. Модель намагається сконцентрувати майже всю масу ймовірності на одному виході – тому самому, що і цільова one-hot мітка. Дана функція зарекомендувала себе як надійний критерій для задач класифікації і широко використовується в сучасних нейронних мережах. Якщо кількість класів $K = 2$ (наприклад, лише позитивний vs негативний), часто застосовується спрощена версія – бінарна крос-ентропія, яка обчислюється для однієї вихідної логістичної активації (англ. sigmoid). Однак у випадку трьох і більше тональних класів природніше використати softmax з багатокласовою крос-ентропією.

Категоріальні тональні мітки формуються зазвичай вручну або краудсорсингом: людські анотатори присвоюють кожному текстовому фрагменту відповідну тональність. Інколи (особливо для нейтрального класу) можливі розбіжності між анотаторами, тоді як остаточну мітку беруть за більшість або середнє значення. В навчальному датасеті тональність може зберігатися як окрема колонка (приміром, текст через табуляцію відділений від мітки positive/negative/neutral).

При навчанні нейромережі останній шар класифікатора видає вектор з K компонент – логіти, що відповідають оцінкам приналежності до кожного класу. Після softmax ці оцінки перетворюються на псевдо-ймовірності p_i (невід’ємні та такі, що $\sum_{i=1}^K p_i = 1$). Під час оптимізації ми порівнюємо цей вихід з one-hot вектором $(y_1, \dots, y_K)$, де $y_c = 1$ для правильного класу c . Втрата

обчислюється як $L = -\sum_{i=1}^K y_i \log p_i$, і зводиться до $-\log p_c$, оскільки $y_i = 0$ для $i \neq c$. Таким чином, мінімізуючи $-\log p_c$, ми максимізуємо ймовірність моделі для правильного класу. На практиці це означає, що модель прагне віддавати $\sim 100\%$ ймовірності правильній мітці, що й потрібно для впевненого класифікатора.

На відміну від одновимірної тональності, яка зводиться до полярності, емоційне забарвлення тексту описується в кількох вимірах одночасно. У розд. 2 було введено поняття вектору емоцій $E(T) = (e_1, e_2, \dots, e_k)$ – де кожна компонента відповідає інтенсивності окремої базової емоції. Зазвичай за базові беруть дискретний набір емоцій (наприклад, 6 емоцій Екмана: радість, сум, гнів, страх, здивування, відраза, + іноді зневага). Відповідно, $k = 6$ або 7 , і вектор може виглядати як, наприклад, $(0.8, 0.0, 0.1, 0.0, 0.6, 0.0)$ для коментаря, що виражає одночасно радість (0.8) і здивування (0.6), з невеликою домішкою гніву (0.1) та відсутністю інших емоцій. Важливо, що тут немає взаємного виключення між класами: один і той самий фрагмент може належати до кількох емоційних категорій одночасно (радісний і здивований), або ж не мати яскраво вираженої жодної емоції (всі компоненти низькі або нульові). Тому таку задачу відносять до багатолейблової класифікації або навіть регресії, якщо інтенсивності трактуються як неперервні числа.

Формування багатовимірних емоційних міток – більш складний процес, ніж тональність. Часто він здійснюється шляхом масштабування або нормування анотацій кількох експертів [84]. Таким чином, у файлі даних багатовимірні мітки можуть бути представлені у вигляді списку емоцій через кому для кожного тексту або як бінарний вектор: 1/0 по кожній з k емоцій. У випадку континуальних анотацій (наприклад, коли кожен емоцію оцінюють за шкалою від 0 до 1 і беруть середнє) отримуємо дійсні значення. Але і їх зазвичай зручно інтерпретувати як «ймовірність» або ступінь присутності відповідної емоції. З точки зору навчання НММ, багатовимірний емоційний вектор – це k вихідних нейронів, кожен з яких відповідає за свою емоцію. Кінцева активація тут, на відміну від тональності, *sigmoid* для кожного виходу

окремо, а не softmax спільно. Це означає, що модель оцінює незалежну ймовірність наявності кожної емоції у фрагменті. Для порівняння з правильними мітками використовується бінарна функція крос-ентропії (2.4) окремо по кожній компоненті виходу.

Інтуїтивно, таким чином модель незалежно вчиться по кожному емоційному виміру відповідати на питання «чи присутня ця емоція?». Такий підхід виправданий, оскільки емоції не виключають одна одну і можуть збігатися.

Використання саме крос-ентропії, замість середньоквадратичної помилки, мотивоване тим, що вирішується k незалежних задач класифікації «емоція vs немає емоції». Бінарна крос-ентропія з логітами є стандартним вибором для мульти-міткових задач, оскільки вона дозволяє моделювати кожну категорію пробабілістично незалежно

Дослідження підтверджують, що така функція втрат добре працює навіть при великій кількості етикеток і дисбалансі між ними. За потреби можуть додатково застосовуватися вагові коефіцієнти або модифікації на кшталт Focal Loss для впорядкування рідкісних емоцій, але базовим є саме BCE.

Припустімо, модель проаналізувала речення і видала на виході вектор оцінок [0.81, 0.05, 0.12, 0.03, 0.67, 0.10] для [радість, сум, гнів, страх, здивування, відроза] відповідно. Якщо поріг ухвалення для класифікації встановлено 0.5, то модель класифікує емоції цього речення як радість і здивування. Це узгоджується з інтуїтивним трактуванням значень >0.5 як «емоція присутня». У випадку континуальних анотацій інтерпретація така ж, просто пороги менш очевидні (можна порівнювати з середнім фоном або використати ранговий підхід). Для оптимізації моделі не потрібно явних порогів – вона безперервно оновлює ваги, щоби збільшити p_i для тих емоцій, де $y_i = 1$, і зменшити для тих, де $y_i = 0$. Варто підкреслити, що у багатовимірному випадку модель може одночасно передбачати кілька «позитивних» класів []. Це складніше завдання, ніж проста тональність, бо емоційні стани часто переплітаються. Важливо також, що нейтральний стан

тут не представлено окремим класом – він радше відповідає ситуації, коли всі e_i близькі до 0 (відсутність виражених емоцій). Тому при підготовці навчального набору бажано мати достатньо прикладів як чисто нейтральних текстів (для навчання моделі видавати нулі), так і різних комбінацій емоцій.

Функція втрат *BCE* добре підходить для цього, бо не вимагає конкуренції між класами: модель може вільно «активувати» стільки виходів, скільки потрібно. Приклад розмітки і формат, представлено в таблиці 3.1.

Таблиця 3.1

Приклади формування датасету, для навчальних і тестових вибірок

N	Текст	Мітки
1	Я здивований і радий одночасно.	joy=1; surprise=1; sadness=0; anger=0; fear=0; disgust=0
2	Мені страшно і сумно.	joy=0; surprise=0; sadness=1; anger=0; fear=1; disgust=0

Тобто кожному тексту відповідає набір бінарних індикаторів по k емоціях. У зручному для моделі форматі це можна представити як рядок з k чисел (через кому або пробіл) після тексту, або як окремі стовпці таблиці. Під час навчання ці мітки будуть використані як цільові значення y_i . З точки зору реалізації, багато фреймворків вимагають просто подавати масив у розміру $(batch \times k)$. Такий підхід підтверджено ефективним для мульти-класифікаційних задач з можливістю належності до кількох класів.

3.2 Аналітичне забезпечення методу

Для забезпечення системності, відтворюваності та теоретичної обґрунтованості запропонованого підходу, перед описом покрокових етапів необхідно формалізувати розроблений метод у вигляді строгої математичної моделі. Ця модель визначає процес розпізнавання емоцій як послідовність функціональних перетворень, що відображають вхідний текстовий фрагмент у

вихідний багатовимірний простір емоційних інтенсивностей. Такий підхід дозволяє чітко розмежувати логіку методу, яка є інваріантною, від його конкретної імплементації, яка може варіюватися залежно від обраних архітектурних рішень та технологічного стеку. Формалізація слугує концептуальним каркасом, що визначає вхідні та вихідні простори, декомпозицію задачі на функціональні блоки та критерій оптимізації, який керує процесом навчання.

Першим кроком у формалізації є точне визначення вхідних та вихідних даних, а також самої задачі розпізнавання як математичного відображення. Нехай $\mathcal{T}$ позначає простір усіх можливих текстових фрагментів українською мовою. Кожен елемент $T \in \mathcal{T}$ являє собою послідовність токенів скінченної довжини n : $T = (w_1, w_2, \dots, w_n)$, де кожен токен w_i належить до фіксованого, але потенційно відкритого словника V . Цей словник включає не лише лексичні одиниці (слова, підслова), але й спеціальні службові токени, такі як $\langle UNK \rangle$ для невідомих слів, $\langle PAD \rangle$ для вирівнювання довжини послідовностей, а також маркери початку та кінця послідовності.

Цільовим простором є простір емоційних станів. Нехай $\mathcal{E} = \{e_1, e_2, \dots, e_k\}$ є фіксованою множиною з k базових емоцій. Оскільки метод спрямований на кількісну оцінку інтенсивності кожної емоції, а не на вибір однієї домінуючої, вихідний простір $\mathcal{Y}$ визначається як k -вимірний гіперкуб. Кожна точка в цьому просторі є вектором інтенсивностей, де кожна компонента може приймати будь-яке значення від 0 (повна відсутність емоції) до 1 (максимальна присутність). Таким чином, $\mathcal{Y} = \mathcal{E}^k$.

Запропонований метод M формально визначається як функція (відображення), що ставить у відповідність кожному текстовому фрагменту з простору $\mathcal{T}$ унікальний вектор емоційних інтенсивностей з простору $\mathcal{Y}$:

$$M: \mathcal{T} \rightarrow \mathcal{Y} \quad (3.2)$$

Для довільного текстового фрагмента $T = (w_1, w_2, \dots, w_n) \in \mathcal{T}$, метод

повертає вектор $y \in \mathcal{Y}$:

$$y = M(T) = (y_1, y_2, \dots, y_k), \quad (3.3)$$

де кожна компонента y_i представляє прогнозовану інтенсивність i -ї базової емоції $e_i \in \mathcal{E}$.

Оскільки функція M не є відомою апріорі, вона має бути навчена на основі даних. Процес навчання використовує анотований набір даних:

$$\mathcal{D} = \{(T_j, y_j^{\text{true}})\}_{j=1}^N, \quad (3.4)$$

де N - кількість прикладів, а y_j^{true} - еталонний вектор емоцій, наданий експертами. Метод M є параметризованим сімейством функцій, де конкретна функція визначається вектором параметрів $\theta \in \Theta$.

Таким чином, задача навчання зводиться до пошуку оптимального набору параметрів θ^* , який найкращим чином апроксимує невідому залежність на основі наявних даних $\mathcal{D}$. Складність відображення M та необхідність забезпечення модульності й інтерпретованості диктують доцільність його декомпозиції на послідовність функціональних блоків. Кожен блок виконує чітко визначене перетворення даних, передаючи результат наступному. Таким чином, метод M представляється у вигляді композиції трьох основних функцій:

$$M = C \circ F \circ E \quad (3.5)$$

де E – функція представлення (Embedding Function); F – функція вилучення контекстуальних ознак (Feature Extraction Function); C – функція класифікації; $\circ$ – позначає оператор функціональної композиції.

Тому:

$$M(T) = C(F(E(T))) \quad (3.6)$$

Така структура дозволяє незалежно аналізувати, реалізовувати та оптимізувати кожен етап процесу розпізнавання.

Перший блок $(E(T))$ відповідає за перетворення дискретних, символьних вхідних даних у неперервний числовий простір, придатний для обробки нейронними мережами. Його головне завдання - початкове семантичне кодування.

Формально, функція E відображає вхідну послідовність tokenів T довжиною n у матрицю векторних представлень (вкладень) X розмірністю $n \times d$, де d - розмірність простору вкладень.

$$E: T \rightarrow X, \quad \text{де } X \in R^{n \times d} \quad (3.7)$$

Кожен рядок x_i матриці X є d -вимірним вектором, що відповідає i -му токеноу w_i . Цей блок є фундаментальним, оскільки якість початкових векторних представлень визначає обсяг семантичної інформації, доступної для наступних етапів.

Другий блок $(F(\dots))$ є обчислювальним ядром методу. Він приймає на вхід матрицю початкових, переважно контекстно-незалежних, представлень X і перетворює її на матрицю H збагачених, контекстуалізованих ознак. Завдання цього блоку - змодельовати складні синтаксичні та семантичні залежності між токенами у послідовності, враховуючи їхній локальний та глобальний контекст.

$$F: X \rightarrow H, \quad \text{де } H \in R^{n \times m} \quad (3.8)$$

Матриця H має розмірність $n \times m$, де m - розмірність простору прихованих станів. Кожен вектор-рядок h_i цієї матриці є представленням

токена w_i , що враховує його взаємодію з іншими токенами у фрагменті T . Цей блок відповідає за "глибоке розуміння" тексту.

Фінальний блок приймає на вхід матрицю контекстуальних ознак H і виконує два послідовні завдання: агрегацію інформації та власне класифікацію. Тому:

$$C = C_{\text{classify}} \circ C_{\text{aggregate}}, \quad (3.9)$$

$$C_{\text{aggregate}} : R^{n \times m} \rightarrow R^{m'} \quad (3.10)$$

Функція агрегації $C_{\text{aggregate}}$ стискає інформацію з усієї послідовності n векторів у єдиний вектор фіксованої розмірності $\mathbf{h}_{agg} \in R^{m'}$, який репрезентує весь текстовий фрагмент.

$$C_{\text{classify}} : R^{m'} \rightarrow R^k \quad (3.11)$$

Функція класифікації C_{classify} приймає цей агрегований вектор $\mathbf{h}_{agg}$ і відображає його у вихідний простір емоцій $\mathcal{Y}$, генеруючи фінальний вектор прогнозованих інтенсивностей y .

Процес навчання методу M є задачею оптимізації, керованою цільовою функцією, яка кількісно оцінює якість його роботи на навчальних даних. Метою є знаходження такого набору параметрів θ^* , який мінімізує очікувану помилку на всьому розподілі даних. На практиці це зводиться до мінімізації усередненої функції втрат $\mathcal{L}(\theta)$ на навчальному наборі $\mathcal{D}$:

$$\theta^* = \arg \min_{\theta \in \Theta} \mathcal{L}(\theta) = \arg \min_{\theta \in \Theta} \frac{1}{N} \sum_{j=1}^N L(\mathbf{y}_j^{\text{tu}}, M_{\theta}(T_j)) \quad (3.12)$$

де L - функція втрат, що обчислюється для одного прикладу. Вона вимірює розбіжність між еталонним вектором $\mathbf{y}^{\text{true}}$ та прогнозованим вектором $\hat{y} = M_{\theta}(T)$.

Як було обґрунтовано раніше, для задачі багатоміткової класифікації, де кожна емоція розглядається як незалежна бінарна подія, найбільш придатною є бінарна перехресна ентропія. Загальна функція втрат для одного прикладу обчислюється як сума (або середнє) значень BCE по всіх k емоціях:

$$\begin{aligned} L(\mathbf{y}^{true}, \hat{\mathbf{y}}) &= \sum_{i=1}^k L_{BCE}(y_i^{true}, \hat{y}_i) \\ &= - \sum_{i=1}^k [y_i^{true} \log(\hat{y}_i) + (1 - y_i^{true}) \log(1 - \hat{y}_i)] \end{aligned} \quad (3.13)$$

Ця функція втрат ефективно штрафувє модель за впевнені, але невірні прогнози для кожної з емоцій, спрямовуючи процес навчання на одночасну оптимізацію всього вектора інтенсивностей.

Для підвищення узагальнюючої здатності моделі та запобігання перенавчанню, до цільової функції додається регуляризаційний член $\Omega(\theta)$, який штрафувє надмірну складність моделі. Найчастіше використовується L_2 -регуляризація (гребенева регресія), яка представляє собою суму квадратів усіх вагових коефіцієнтів моделі: $\Omega(\theta) = \frac{1}{2} \|\theta\|_2^2$.

Таким чином, фінальний варіант аналітичного представлення задачі навчання НММ має вигляд:

$$\begin{aligned} J(\theta) = \mathcal{L}(\theta) + \lambda \Omega(\theta) &= \frac{1}{N} \sum_{j=1}^N L(\mathbf{y}_j^{true}, M_\theta(T_j)) + \\ &\quad \frac{\lambda}{2} \|\theta\|_2^2, \end{aligned} \quad (3.14)$$

де $\lambda \geq 0$ - гіперпараметр, що контролює силу регуляризації.

Розроблене аналітичне забезпечення (3.1-3.14), яке включає визначення просторів, декомпозицію на функціональні блоки та визначення критерію

оптимізації становить теоретичний базис запропонованого методу, що забезпечує можливість формування опису його етапів.

3.3 Етапи методу розпізнавання емоційної тональності

Після того, як у попередньому підрозділі було представлено формалізовану математичну модель методу, необхідно детально описати послідовність етапів, що реалізують цю модель на практиці. Запропонований метод ґрунтується на теоретичних положеннях, викладених у попередніх розділах, та інтегрує сучасні підходи до побудови систем обробки природної мови, адаптуючи їх для вирішення задачі PET в умовах універсальних комп'ютерних засобів. Загальний процес розпізнавання емоційного забарвлення тексту, згідно з розробленою концептуальною моделлю, можна представити у вигляді послідовності ключових етапів, як це схематично зображено на рис. 3.1.

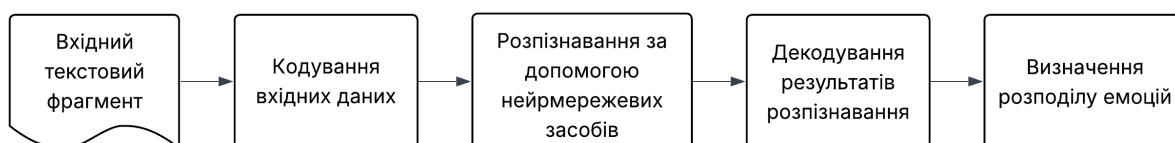


Рис. 3.1. Послідовність дій для процесу розпізнавання емоцій

Важливо наголосити, що хоча логічна послідовність етапів є інваріантною, їхня конкретна технічна реалізація може суттєво відрізнятися. Ця гнучкість є однією з ключових переваг запропонованого модульного методу. Так, етап кодування входу буде реалізований по-різному в залежності від обраної базової архітектури (наприклад, використання статичних Word2Vec ембедингів проти контекстуалізованих представлень трансформера). Аналогічно, етап декодування виходу напряму залежить від способу визначення емоції: якщо задача полягає у визначенні однієї домінуючої емоції з-поміж k взаємовиключних (задача багатокласової класифікації), на виході

буде застосовуватися функція Softmax. Якщо ж, як у даній роботі, визначається ступінь присутності кожної з k емоцій незалежно (задача багатоміткової класифікації), буде використовуватися сигмоїдна функція. Нижче наведено опис етапів для основного випадку, що розглядається в даній дисертації - багатоміткової класифікації.

Розглянемо кожен з цих етапів більш детально, пов'язуючи їх із функціональними блоками математичної моделі, описаної в п. 3.2. Процес ініціалізується подачею на вхід довільного текстового фрагмента $T \in \mathcal{T}$.

Перший етап - кодування входу - є реалізацією функції представлення E з математичної моделі. Його головне завдання - трансформувати сирий, символічний текст у числовий формат (тензор), придатний для обробки нейронною мережею. Цей етап сам по собі є багатокроковим процесом, що включає токенизацію та векторизацію. Спочатку вхідний текст T розбивається на послідовність токенів $(w_1, w_2, \dots, w_n)$ за допомогою обраного токенизатора. Далі, ця послідовність перетворюється на матрицю векторних представлень. Узагальнено цей процес можна описати так:

$$X = E(T) = (E_{\text{embed}} \circ E_{\text{tokenize}})(T), \quad (3.15)$$

де E_{tokenize} - це оператор токенизації, що перетворює рядок на послідовність токенів, а E_{embed} - оператор векторизації, що відображає кожен токен у d -вимірний вектор. Результатом є матриця $X \in R^{n \times d}$, де кожен рядок x_i є векторним представленням i -го токена. Ця матриця є входом для наступного етапу.

Етап розпізнавання нейромережевою моделлю. Вилучення та агрегація ознак. На цьому етапі відбувається безпосередня обробка числового представлення тексту навченою нейромережевою моделлю. Цей крок реалізує композицію функції вилучення контекстуальних ознак F та функції класифікації C . Спочатку модель обробляє вхідну матрицю X для виявлення

прихованих патернів та залежностей, генеруючи матрицю контекстуалізованих ознак H :

$$H = F(X) \quad (3.16)$$

Далі, ця матриця ознак агрегується та подається на класифікаційну "голову" для отримання вихідного вектора логітів. Логіти - це сирі, ненормалізовані виходи моделі перед застосуванням фінальної функції активації:

$$z = C(H) = (z_1, z_2, \dots, z_k). \quad (3.17)$$

Кожен елемент z_i цього вектора відповідає "впевненості" моделі у присутності i -ї емоції. Весь цей етап є прямим проходом (forward pass) через нейронну мережу, де вхідний тензор послідовно трансформується шарами моделі.

Етап декодування виходу – полягає у перетворенні сирих вихідних сигналів моделі (логітів) у фінальний, інтерпретований результат. Оскільки задача полягає у визначенні інтенсивності кожної з k емоцій незалежно, для перетворення вектора логітів z у вектор ймовірностей (або інтенсивностей) у застосовується сигмоїдна функція активації $\sigma(x) = 1/(1 + e^{-x})$ покомпонентно до кожного елемента z :

$$y = \sigma(z) = (\sigma(z_1), \sigma(z_2), \dots, \sigma(z_k)). \quad (3.18)$$

Результатом є розподіл ймовірностей для k емоцій, представлений вектором $y = (y_1, y_2, \dots, y_k)$, де кожна компонента y_i відповідає ймовірності (або ступеню) присутності i -ї емоції в тексті. Важливо, що на відміну від функції Softmax, яка змушує суму ймовірностей дорівнювати одиниці, сигмоїдна функція дозволяє кожній емоції мати свою незалежну ймовірність, що повністю відповідає постановці задачі багатоміткової класифікації.

Центральним елементом запропонованого модульного методу є не лише гнучкість у виборі архітектури, але й наявність формалізованої процедури для здійснення цього вибору. Як було обґрунтовано в розділі 2, різні нейромережеві архітектури демонструють різний компроміс між предиктивною точністю та обчислювальною вартістю. Тому першим кроком у практичній реалізації методу є обґрунтований вибір архітектури НММ оскільки саме він визначає подальшу специфіку процедур обробки даних, вимоги до апаратного забезпечення та фінальні експлуатаційні характеристики системи. Для систематизації цього процесу було розроблено алгоритм, візуалізований на рис. 3.2.

В основі алгоритму лежить аналіз ресурсних обмежень та цільових пріоритетів. Цей підхід дозволяє проектувати рішення, які є не просто теоретично точними, а й практично придатними для розгортання в конкретних умовах універсальних комп'ютерних систем. Нехай $\mathcal{C}$ - це множина обмежень цільової платформи, що включає доступність апаратних прискорювачів (CPU, GPU), ліміти на споживання пам'яті та енергії, а також вимоги до максимальної затримки відповіді (latency).

Нехай $\mathcal{P}$ - це множина пріоритетів, що визначає, яка метрика є ключовою для задачі (наприклад, максимальна точність (F1-міра) або максимальна швидкість (throughput)). Алгоритм реалізує функцію вибору $\mathcal{F}_{arch}$, яка відображає ці множини у конкретний клас архітектури:

$$\text{Архітектура} = \mathcal{F}_{arch}(\mathcal{C}, \mathcal{P}) \quad (3.19)$$

Механізм розрахунку виразу (3.19), зображений на рисунку 3.2. Зазначимо, що вказаний алгоритм є практичною реалізацією процедури максимізації критерію сумарної ефективності E_{st} , визначеної виразом (2.14). Аналіз множини обмежень $\mathcal{C}$ відповідає врахуванню допустимих значень ресурсоємності (E_{nmm}), а множина пріоритетів $\mathcal{P}$ визначає вагові коефіцієнти

у метриці інтегральної якості (Q_{total}). При цьому визначення архітектури передбачає реалізацію двох сценаріїв.

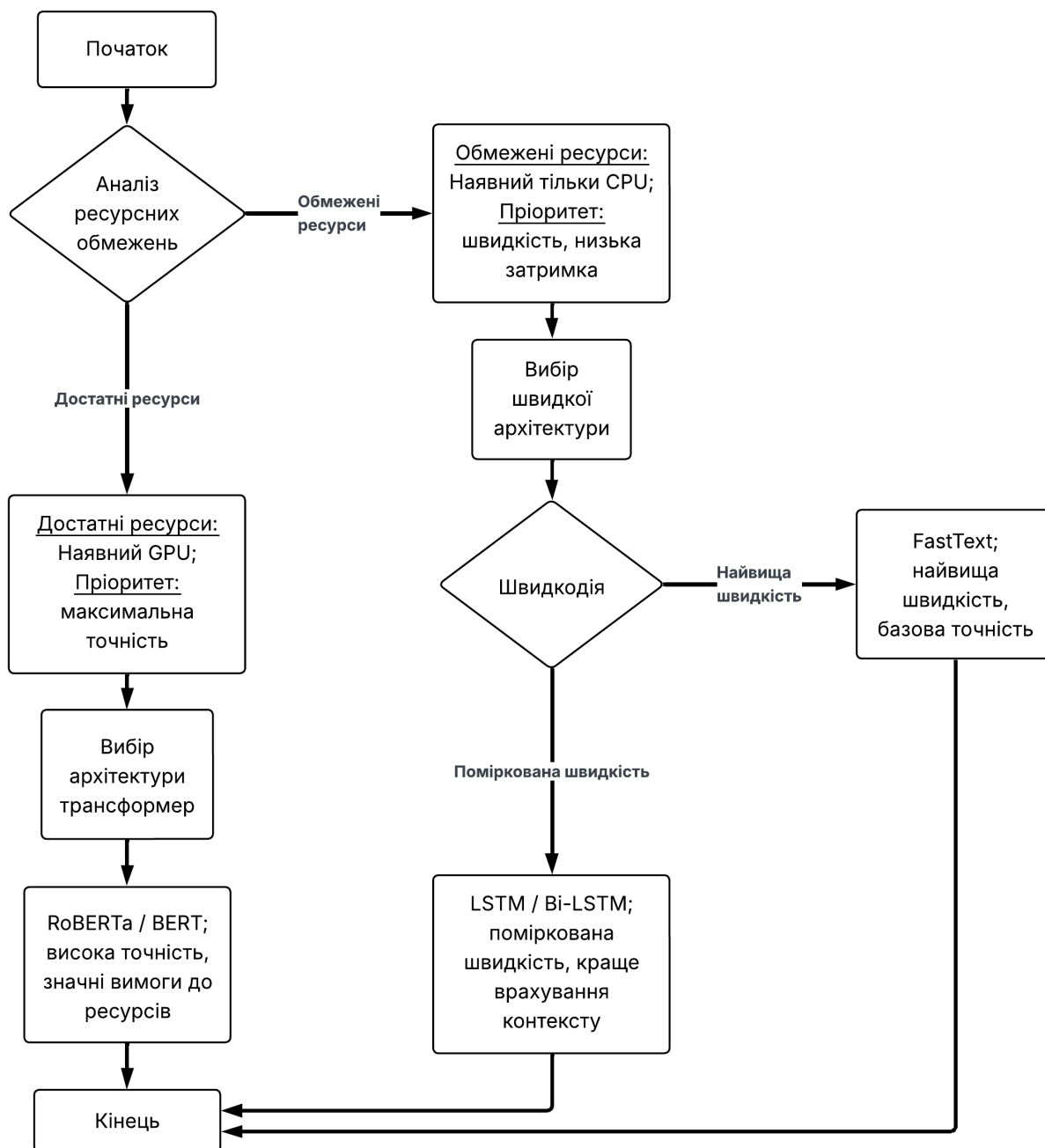


Рис. 3.2. Алгоритм вибору архітектури НММ

Сценарій 1. Обмежені ресурси (Пріоритет: швидкість, низька затримка)

Цей сценарій є типовим для мобільних пристроїв, вбудованих систем, IoT-пристроїв або серверних застосунків, де обробка відбувається виключно на центральному процесорі (англ. Central Processing Unit, CPU), а вимоги до швидкості реакції та енергоефективності є критичними. В таких умовах

пріоритет надається швидким архітектурам, які мають низьку обчислювальну складність та малий об'єм пам'яті.

Якщо ключовою вимогою є мінімально можлива затримка та найменше споживання ресурсів (наприклад, для попередньої фільтрації великих потоків даних), оптимальним вибором є архітектура FastText. Вона базується на усередненні векторних представлень слів та n -грам символів, що дозволяє досягти надзвичайно високої швидкості інференсу. Її обчислювальна складність є лінійною відносно довжини тексту, $O(n \cdot d)$, що робить її ідеальною для CPU-інтенсивних завдань, хоча й ціною спрощеного врахування контексту.

Якщо апаратні ресурси CPU дозволяють більш об'ємні обчислення, а для задачі є важливим врахування порядку слів та локального контексту, кращим вибором є архітектура на основі LSTM/Bi-LSTM. Хоча її рекурентна природа робить її повільнішою за FastText (обчислювальна складність $O(n \cdot d^2)$), вона здатна моделювати послідовні залежності, що зазвичай призводить до значного підвищення точності. Цей варіант є збалансованим рішенням для багатьох серверних застосунків, що працюють на CPU.

Сценарій 2. Достатні ресурси (Пріоритет: максимальна точність)

Цей сценарій є типовим для систем, де точність розпізнавання має найвищий пріоритет, а для обчислень доступні графічні процесори (англ. Graphics Processing Unit, GPU) або інші апаратні прискорювачі. Це можуть бути системи для глибокого офлайн-аналізу, виявлення дезінформації або будь-які інші задачі, де ціна помилки є високою. В цьому випадку алгоритм однозначно вказує на вибір архітектури трансформер. Завдяки механізму самоуваги, обчислювальна складність якого становить $O(n^2 \cdot d)$, трансформери здатні моделювати складні, довготривалі та нелінійні залежності між усіма токенами в тексті. Це дозволяє їм досягати найвищих показників точності. В рамках цього вибору найчастіше використовуються попередньо навчені моделі, такі як RoBERTa, BERT або їх багатомовні варіанти. Ці моделі вже містять у собі величезний обсяг лінгвістичних знань, отриманих під час

попереднього навчання на гігантських текстових корпусах, і потребують лише відносно невеликого донавчання на цільовій задачі. Хоча їхні вимоги до ресурсів (як пам'яті, так і обчислювальної потужності) є значними, для задач з пріоритетом максимальної точності та наявністю GPU вони є безальтернативним вибором.

Після визначення оптимального класу архітектури нейромережевої моделі згідно з алгоритмом, представленим у попередньому підрозділі 3.2, наступним критичним етапом запропонованого методу є її навчання. Цей процес є центральним у життєвому циклі моделі, оскільки саме під час нього абстрактна, неініціалізована архітектура перетворюється на функціональний інструмент, здатний розв'язувати поставлену задачу розпізнавання емоцій. Процес навчання не є довільним набором евристик; він реалізується згідно зі строгою, структурованою методологією, теоретичні основи якої були закладені в попередніх розділах.

Зокрема, загальна послідовність дій та логічний взаємозв'язок етапів регламентуються концептуальною моделлю, детально описаною в розділі 2. Ця модель забезпечує макрорівневий погляд на процес, визначаючи місце навчання в загальному конвеєрі від підготовки даних до фінальної валідації. Практичні аспекти реалізації, що включають формування навчальних та валідаційних вибірок, вибір оптимізаційного алгоритму, а також налаштування гіперпараметрів, ґрунтуються на математичній моделі методу, формально описаній у підрозділі 3.2, та деталізованих архітектурних рішеннях з підрозділу 2.3. Таке поєднання теоретичного каркасу та практичної методології забезпечує системність, відтворюваність та керованість усього процесу навчання.

Процедура навчання є ітеративним процесом оптимізації, метою якого є знаходження такого набору параметрів (вагових коефіцієнтів) θ^* моделі M , який мінімізує значення функції втрат $J(\theta)$ на навчальному наборі даних. Цей процес складається з кількох взаємопов'язаних кроків, що циклічно

повторюються протягом багатьох епох: від подачі на вхід моделі порції даних (міні-батча) до коригування її ваг на основі обчисленої помилки.

Специфіка реалізації етапу кодування залежить від обраної архітектури нейронної мережі. Розглянемо їх для трьох основних класів моделей.

1. FastText (та подібні представники)

Процес кодування для архітектури FastText, як показано на рис. 3.3, є послідовним перетворенням вхідного тексту на єдиний вектор фіксованої розмірності. Він включає наступні кроки:

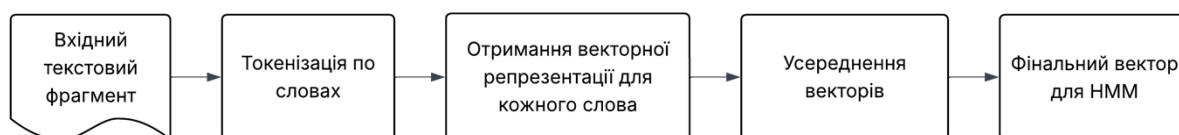


Рис. 3.3. Етапи кодування вхідних даних для архітектури FastText

Токенізація за словами: На першому кроці вхідний текстовий фрагмент T піддається токенизації на рівні слів, перетворюючись на послідовність $T_{tok} = (w_1, w_2, \dots, w_n)$.

Отримання векторів слів з урахуванням підслівних ознак: На відміну від класичного Word2Vec, ключовою особливістю FastText є представлення кожного слова як "мішка" символьних n -грам. Це дозволяє генерувати вектори навіть для слів, відсутніх у словнику (OOV). Вектор v_{w_i} для кожного слова w_i обчислюється як сума векторів його символьних n -грам та вектора самого слова (якщо воно є у словнику). Формально, нехай $G(w_i)$ - це множина символьних n -грам, що складають слово w_i . Тоді його векторне представлення:

$$v_{w_i} = \frac{1}{|G(w_i)| + 1} \left(v'_{w_i} + \sum_{g \in G(w_i)} \text{bz}_g \right) \quad (3.20)$$

де v'_{w_i} - вектор самого слова, а z_g - вектор, що відповідає n -грамі g . Цей підхід забезпечує робастність до друкарських помилок та морфологічної варіативності.

Агрегація векторів: Для отримання єдиного вектора, що характеризує весь текстовий фрагмент, застосовується усереднення векторів усіх його токенів. Цей підхід, відомий як "Bag-of-Embeddings", є реалізацією функції агрегації $C_{\text{aggregate}}$. Фінальний вектор h_{agg} , що репрезентує текст, обчислюється як:

$$h_{agg} = \frac{1}{n} \sum_{i=1}^n v_{w_i} \quad (3.21)$$

Цей вектор $h_{agg} \in R^d$ і є результатом роботи етапу кодування, який подається на вхід нейромережевої моделі.

Реалізація розпізнавання нейромережею для архітектури FastText виконує функція вилучення контекстуальних ознак F і є тотожною C , оскільки підхід "Bag-of-Embeddings" ігнорує порядок слів і, відповідно, складний контекст. Агрегований вектор h_{agg} напряду подається на вхід класифікаційної "голови", що реалізує функцію C_{classify} .

Ця "голова" зазвичай є простою нейронною мережею, що складається з одного або декількох повнозв'язних шарів. Наприклад, для одного прихованого шару процес обчислення логітів z можна описати так:

$$z = W_2 \cdot \text{ReLU}(W_1 h_{agg} + b_1) + b_2 \quad (3.22)$$

де W_1, b_1, W_2, b_2 - матриці ваг та вектори зсувів, що навчаються, а ReLU - функція активації. Отриманий вектор логітів $z \in R^k$ далі перетворюється на фінальний розподіл ймовірностей (інтенсивностей) y за допомогою покомпонентної сигмоїдної функції, як це було описано у формулі (3.4).

2. LSTM/Bi-LSTM

На відміну від підходу, що використовується у FastText, рекурентні мережі обробляють текст як впорядковану послідовність, де порядок слів має ключове значення. Етап кодування, зображений на рис. 3.4, спрямований на перетворення тексту в формат, що зберігає цю послідовну природу.

Токенізація: Як і в попередньому випадку, вхідний текст T спочатку токенизується, перетворюючись на послідовність токенів $T_{tok} = (w_1, w_2, \dots, w_n)$

Отримання послідовності векторів: Кожен токен w_i з послідовності відображається у відповідний йому d -вимірний вектор x_i за допомогою шару вкладень (embedding layer). На відміну від FastText, на цьому етапі агрегація не відбувається. Результатом роботи функції представлення $E(T)$ є повна послідовність векторів, яка зберігає порядок слів і може бути представлена у вигляді матриці X : $X = (x_1, x_2, \dots, x_n)^T \in R^{n \times d}$. Ця матриця X подається на вхід рекурентних шарів для подальшої обробки.

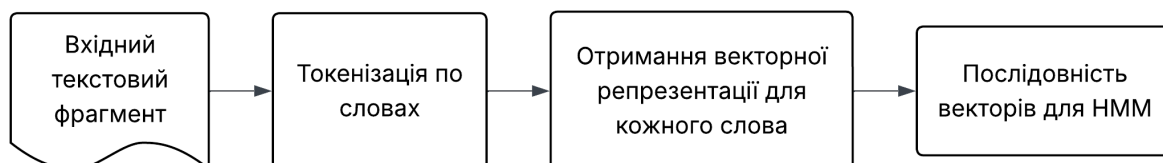


Рис. 3.4. Етапи кодування вхідних даних для архітектури LSTM

На етапі розпізнавання HMM (реалізація функції вилучення ознак $F(X)$ та класифікації $C(X)$) реалізується ключова перевага рекурентних архітектур - моделювання контекстуальних зв'язків та залежностей між словами. Матриця вхідних векторів X послідовно обробляється одним або декількома шарами LSTM (або Bi-LSTM). Для кожного токена x_t на часовому кроці t , LSTM-комірка оновлює свій прихований стан h_t та стан комірки c_t , акумулюючи інформацію з попередніх кроків, як це було детально описано у формулі (2.16). Результатом роботи цього блоку є послідовність прихованих станів:

$$H = (h_1, h_2, \dots, h_n) = F(X) \in R^{n \times m}, \quad (3.23)$$

де m - розмірність прихованого стану LSTM.

Кожен вектор h_t у цій послідовності є представленням токена w_t , що враховує весь попередній контекст. У випадку використання Bi-LSTM, кожен вектор h_t є конкатенацією прихованих станів прямого $\vec{h}_t$ та зворотного $\overleftarrow{h}_t$ проходів, що дозволяє враховувати як попередній, так і наступний контекст.

Після отримання матриці контекстуалізованих прихованих станів необхідно агрегувати цю інформацію для отримання єдиної репрезентації всього текстового фрагмента h_{agg} . Отриманий агрегований вектор $h_{agg} \in R^m$ подається на класифікаційну "голову", яка, аналогічно до попереднього випадку, складається з одного або декількох повнозв'язних шарів (3.22) для генерації фінального вектора логітів z . Подальше перетворення логітів у розподіл ймовірностей відбувається за допомогою етапу декодування.

3. Transformer

HMM на основі архітектури Трансформер використовують найбільш досконалий на сьогодні підхід до кодування тексту, що забезпечує робастність та високу семантичну виразність. Процес, зображений на рис. 3.5, є більш складним порівняно з попередніми архітектурами.

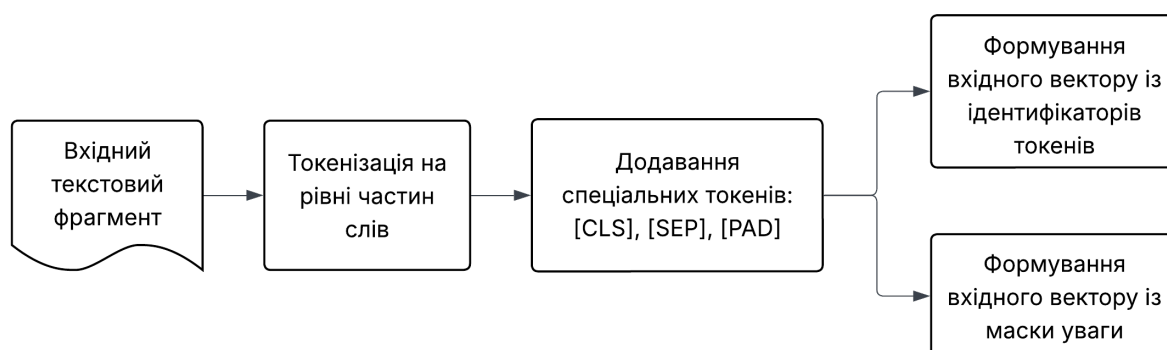


Рис. 3.5. Етапи кодування вхідних даних для архітектури Transformer

Він складається з наступних кроків:

Токенізація на рівні частин слів. На відміну від токенизації за словами, тут застосовується гранулярний підхід, відомий як підслівна-токенізація (англ. subword-tokenization, напр., WordPiece, SentencePiece). Вхідний текст T розбивається на послідовність підслівних одиниць $T_{sub} = (w_1, w_2, \dots, w_n)$, що дозволяє ефективно працювати з рідкісними, невідомими словами та складною морфологією, практично повністю вирішуючи проблему OOV.

Додавання спеціальних токенів. До отриманої послідовності додаються службові токени, що є невід'ємною частиною архітектури BERT-подібних моделей. Зазвичай на початок послідовності додається токен $[CLS]$, а в кінець - $[SEP]$. Таким чином, фінальна послідовність має вигляд $T_{final} = (w_{[CLS]}, w_1, \dots, w_n, w_{[SEP]})$.

Формування вхідних тензорів. На вхід моделі подаються не безпосередньо вектори, а кілька тензорів, що містять всю необхідну інформацію:

Ідентифікатори токенів. Це тензор цілих чисел, де кожен токен з T_{final} замінено на його унікальний індекс зі словника моделі: $I = (i_{[CLS]}, i_1, \dots, i_n, i_{[SEP]})$.

Маска уваги. Це бінарний тензор A тієї ж довжини, що й I , який вказує моделі, на які токени слід звертати увагу. Він містить «1» для реальних токенів і «0» для спеціальних «<PAD>» токенів, що використовуються для вирівнювання довжини послідовностей у батчі. Ці тензори, I та A , є результатом роботи етапу кодування і подаються на вхід трансформерної моделі.

Етап розпізнавання НММ (функції вилучення ознак $F(X)$ та класифікації $C(X)$) реалізується за допомогою послідовності шарів кодувальника трансформера. Модель приймає на вхід ідентифікатори токенів, перетворює їх на вектори (додаючи позиційні та сегментні вкладення) і послідовно обробляє їх за допомогою механізму багатоголової самоуваги, як це було описано у

формулі (2.17). Результатом роботи цього блоку є матриця контекстуалізованих прихованих станів $H \in R^{(n+2) \times m}$:

$$H = F(I, A) = (h_{[CLS]}, h_1, \dots, h_n, h_{[SEP]}) \quad (3.24)$$

Кожен вектор h_i у цій матриці є глибоко контекстуалізованим представленням відповідного токена, що враховує його взаємозв'язки з усіма іншими токенами в послідовності. Для задачі класифікації всього тексту стандартною практикою для BERT-подібних архітектур є використання векторного стану, що відповідає службовому токenu '[CLS]'. Вважається, що завдяки механізму само-уваги, прихований стан $h_{[CLS]}$ після проходження через усі шари агрегує в собі інформацію про всю послідовність. Таким чином, функція агрегації $C_{\text{aggregate}}$ зводиться до вибору цього вектора: $h_{\text{agg}} = h_{[CLS]}$.

Цей агрегований вектор $h_{\text{agg}} \in R^m$ подається на класифікаційну "голову", яка зазвичай складається з одного повнозв'язного шару (3.22).

Незалежно від обраної архітектури – FastText, LSTM Трансформер – фінальний етап методу є декодування виходу. Цей етап є концептуально уніфікованим і відповідає за перетворення внутрішніх, неструктурованих вихідних даних моделі у фінальний, інтерпретований результат - ймовірнісну оцінку емоційного стану тексту. Останній повнозв'язний шар будь-якої з розглянутих НММ генерує вектор необроблених числових значень – логіти. Ці значення, $z = (z_1, z_2, \dots, z_k)$, представляють собою ненормалізовану "впевненість" моделі у присутності кожної з k емоцій. Для перетворення цього вектора у значущий розподіл ймовірностей застосовується відповідна функція активації, вибір якої критично залежить від постановки задачі.

Як показано на рис. 3.6, для задачі багатокласової класифікації, де необхідно визначити одну домінуючу емоцію з-поміж k взаємовиключних класів, застосовується функція – Softmax.

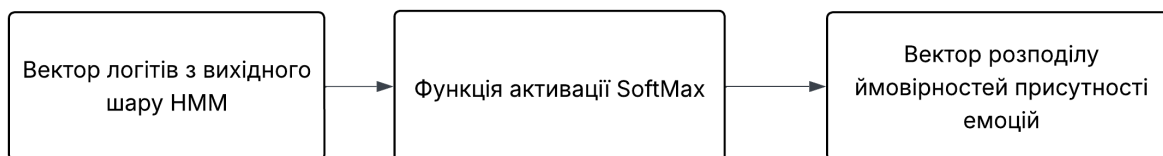


Рис. 3.6. Етапи декодування виходу НММ

Вона нормалізує вектор логітів таким чином, що кожен елемент вихідного вектора знаходиться в діапазоні $[0, \dots, 1]$, а сума всіх елементів дорівнює 1. Ймовірність p_i для i -го класу обчислюється за формулою:

$$p_i = \text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^k e^{z_j}}. \quad (3.25)$$

Виходячи із (3.25), вихідний вектор $p = (p_1, \dots, p_k)$ можна інтерпретувати як розподіл ймовірностей належності тексту до одного з емоційних класів. Однак, як було обґрунтовано в даній дисертації, такий підхід є обмеженим. Тому для основного завдання - багатоміткової класифікації, де визначається ступінь присутності кожної емоції незалежно, - застосовується функція Sigmoid, як це було детально описано у формулі (3.18). Цей вибір дозволяє моделі розпізнавати складні емоційні стани, де в тексті одночасно присутні кілька емоцій.

Таким чином, запропонований метод розпізнавання емоційної тональності є комплексним, багатоетапним процесом, що охоплює весь шлях від обробки сирого тексту до отримання фінального, інтерпретованого результату. Було представлено гнучкий алгоритм вибору оптимальної архітектури, що дозволяє адаптувати метод до різних ресурсних обмежень та вимог до точності. Для кожного класу архітектур - від легкої FastText до потужних Трансформерів - було детально описано та формалізовано процедури кодування, розпізнавання та декодування, що реалізують єдину математичну модель, розроблену в підрозділі 3.2. Цей системний та модульний

підхід забезпечує не лише високу ефективність розпізнавання, але й відтворюваність та адаптивність методу, що є ключовими вимогами для його впровадження в універсальних комп'ютерних системах.

3.4. Висновки до третього розділу

Третій розділ дисертаційної роботи присвячено вирішенню наукового завдання, що полягає у розробці методу розпізнавання емоційної тональності текстових фрагментів з використанням нейромережевих засобів в універсальних комп'ютерних системах. При цьому отримано такі наукові результати:

- Обґрунтовано процедуру підготовки вхідних даних, яка базується на використанні токенізації на основі підслівних одиниць. Це забезпечило вирішення проблеми обробки слів, відсутніх у словнику, та врахування складної морфології української мови, що стало фундаментом для розширення лексичного покриття засобів розпізнавання.

- Розроблено математичну модель, яка формалізує процес розпізнавання як композицію функцій представлення, вилучення контекстуальних ознак та класифікації. Така декомпозиція створила теоретичне підґрунтя для гнучкого конструювання нейромережевих засобів, а визначений критерій оптимізації на основі бінарної перехресної ентропії дозволив реалізувати навчання моделей для задачі багатоміткової класифікації емоцій .

- Розроблено механізм вибору архітектури нейромережевої моделі, що ґрунтується на аналізі ресурсних обмежень цільової платформи та пріоритетів швидкодії або точності. Механізм реалізує процедуру максимізації критерію сумарної ефективності, дозволяючи обґрунтовано обирати між легковаговими та глибокими архітектурами, що забезпечує адаптацію засобів до умов застосування.

Таким чином розроблений метод розпізнавання емоційної тональності фрагментів тексту, за рахунок комплексного використання моделі побудови нейромережевих засобів, удосконаленої системи критеріїв оцінювання та

розробленого анотованого корпусу україномовних текстів, забезпечує можливість адаптації архітектури засобу розпізнавання до умов застосування та розширення лексичного покриття предметної області, що дозволяє підвищити якість розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

РОЗДІЛ 4

СИСТЕМА РОЗПІЗНАВАННЯ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ ФРАГМЕНТІВ ТЕКСТУ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

4.1 Архітектура системи розпізнавання

Для проведення експериментальних досліджень та практичної демонстрації розробленого методу було спроектовано та реалізовано програмну систему у вигляді клієнт-серверного веб-додатку. Цей підрозділ описує архітектуру даної системи, її ключові компоненти та взаємозв'язки між ними з використанням стандартної мови моделювання UML (Unified Modeling Language). Опис архітектури є необхідним для розуміння того, як теоретичний метод, розроблений у розділі 3, був втілений у конкретному програмному артефакті, що використовувався для отримання всіх експериментальних результатів.

Програмна система спроектована за принципами багат шарової архітектури, що забезпечує гнучкість, масштабованість та легкість у модифікації. Архітектуру можна логічно розділити на три основні шари:

- Шар представлення (Presentation Layer): Відповідає за взаємодію з кінцевим користувачем. Реалізований у вигляді веб-інтерфейсу на основі бібліотеки Gradio [85]. Цей шар приймає вхідний текст від користувача, відправляє його на обробку та відображає фінальний результат розпізнавання.
- Шар бізнес-логіки (Business Logic Layer): Є центральним ядром системи, що реалізує логіку розробленого методу. Цей шар отримує запит від шару представлення, оркеструє процес обробки тексту, викликає відповідні моделі та повертає результат.
- Шар доступу до даних та моделей (Data and Model Access Layer): Відповідає за попередню обробку тексту (препроцесинг) та безпосередньо за

виконання розпізнавання за допомогою нейромережевих моделей (інференс). Цей шар інкапсулює складність роботи з токенизаторами та фреймворками глибокого навчання - PyTorch [86]. Для детального опису статичної структури та динамічної поведінки системи було використано діаграми UML.

На рисунку 4.1 представлена діаграма компонентів, що візуалізує основні програмні модулі системи та залежності між ними:

- WebAppInterface: Компонент, що реалізує веб-інтерфейс (шар представлення). Він залежить від EmotionRecognitionService для виконання основної функції програми.
- EmotionRecognitionService: Центральний компонент, що реалізує шар бізнес-логіки. Він діє як фасад, координуючи взаємодію між іншими компонентами.

Діаграма компонентів програмної системи

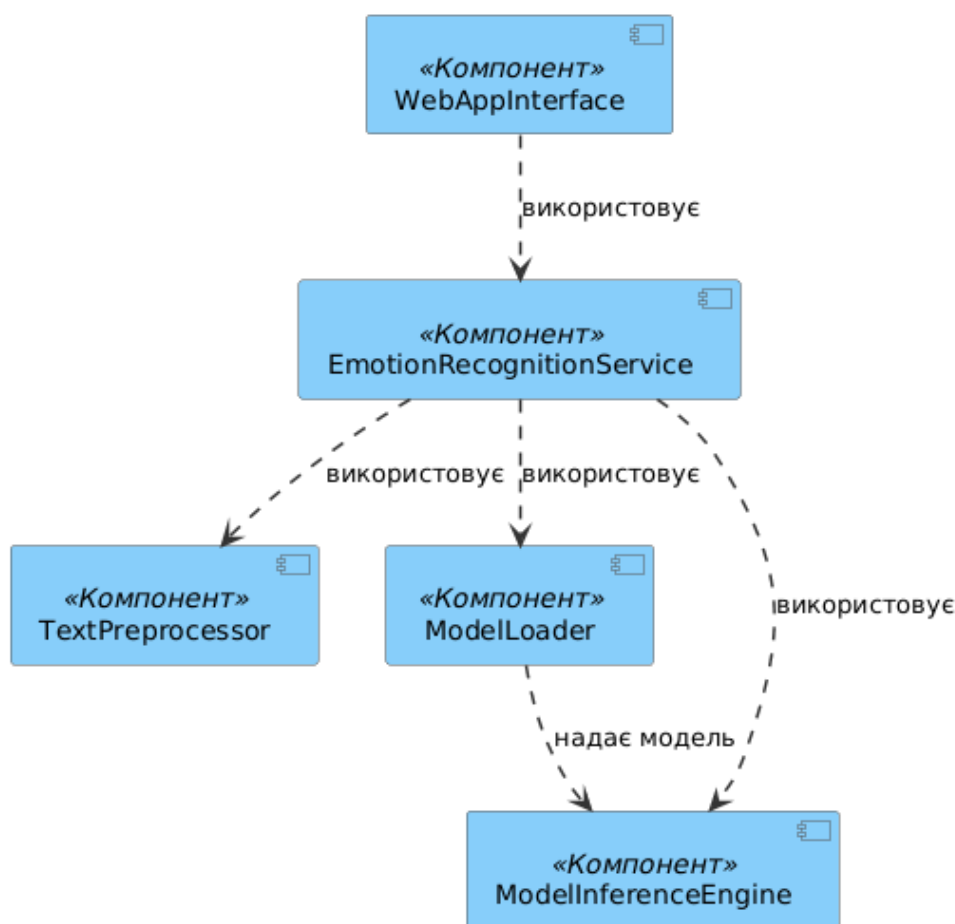


Рис. 4.1. Діаграма компонентів програмної системи

- **TextPreprocessor**: Компонент, відповідальний за всі етапи попередньої обробки тексту: очистку, нормалізацію та токенизацію, специфічну для обраної моделі.

- **ModelLoader**: Компонент, що відповідає за завантаження необхідної нейромережевої моделі (FastText або RoBERTa-base) та її ваг у пам'ять.

- **ModelInferenceEngine**: Компонент, що інкапсулює логіку виконання прямого проходу (inference) через завантажену нейромережеву модель. Він приймає на вхід підготовлені тензори та повертає вектор логітів.

На рис. 4.2 зображено діаграму послідовності, яка демонструє динамічну взаємодію між компонентами системи під час обробки одного запиту від користувача.

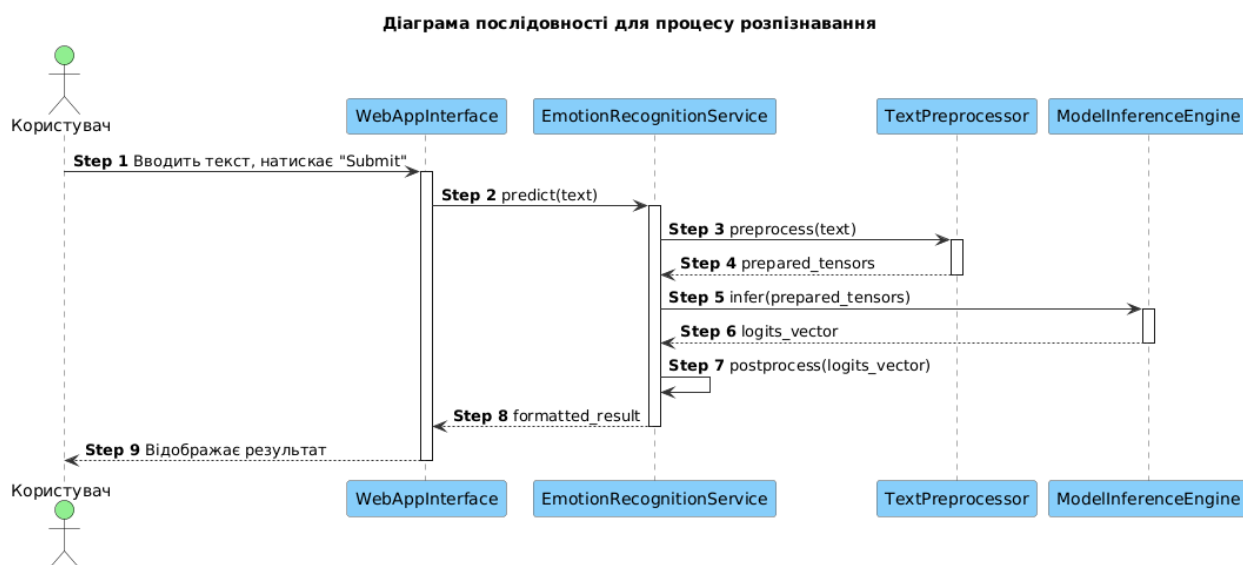


Рис. 4.2. Діаграма послідовності для процесу розпізнавання

Процес розпізнавання відбувається за таким алгоритмом:

1. Користувач (User) вводить текст у веб-інтерфейс та натискає кнопку "Submit".
2. Компонент WebAppInterface отримує текст і викликає метод `predict()` у сервісі EmotionRecognitionService.

3. EmotionRecognitionService спершу передає текст до TextPreprocessor для очистки та токенизації.

4. TextPreprocessor повертає підготовлені вхідні тензори (ідентифікатори токенів та маску уваги).

5. EmotionRecognitionService передає ці тензори до ModelInferenceEngine для виконання розпізнавання.

6. ModelInferenceEngine виконує прямий прохід через нейронну мережу та повертає вектор логітів.

7. EmotionRecognitionService виконує постобробку: застосовує функцію Softmax/Sigmoid до логітів, форматує результат у зрозумілий для користувача вигляд (назва емоції та її ймовірність).

8. Відформатований результат повертається до WebAppInterface, який відображає його користувачеві.

На рисунку 4.3 показано діаграму розгортання, що ілюструє, як програмні компоненти розміщуються на апаратних вузлах у різних експериментальних конфігураціях.

Діаграма демонструє три сценарії розгортання, що відповідають трьом тестовим платформам:

- Сценарій А (Малопотужна система): Усі програмні компоненти розгортаються на єдиному фізичному вузлі Raspberry Pi 4, де обчислення виконуються на ARM CPU.

- Сценарій Б (Десктопна система): Усі компоненти розгортаються на вузлі Apple M3 Pro, де обчислення виконуються на ARM-based CPU.

- Сценарій В (Хмарна система): Архітектура може бути розподіленою. WebAppInterface та EmotionRecognitionService розгортаються на серверному вузлі з Intel Xeon CPU, тоді як найбільш обчислювально затратний компонент ModelInferenceEngine розгортається на окремому вузлі з NVIDIA A10G GPU для максимальної продуктивності.



Рис. 4.3. Діаграма розгортання системи на гетерогенних платформах

Ця архітектурна гнучкість є прямим наслідком розробленого модульного методу та дозволяє ефективно використовувати систему в широкому спектрі універсальних комп'ютерних систем. Описана архітектура є основою для проведення всіх подальших експериментальних досліджень, результати яких представлено в наступних підрозділах.

4.2 Експериментальна установка

Для експериментальної перевірки та валідації розробленого в попередньому розділі методу, а також для оцінки ефективності моделей, що ним генеруються, було створено комплексну експериментальну установку. Цей підрозділ детально описує програмний інструментарій та апаратну інфраструктуру, що використовувалися на всіх етапах дослідження - від збору та підготовки даних до навчання, тестування та інтерпретації нейромережевих моделей. Вибір конкретних інструментів ґрунтувався на критеріях відкритості, відтворюваності, відповідності сучасним стандартам у галузі обробки природної мови та машинного навчання, а також на необхідності симуляції роботи системи в умовах гетерогенних обчислювальних середовищ.

Основою для всієї програмної реалізації було обрано мову програмування Python версії 3.11 [87]. Цей вибір зумовлений її статусом де-факто стандарту в галузі науки про дані та штучного інтелекту, а також наявністю зрілої та всебічної екосистеми бібліотек, що значно прискорює

процес розробки та забезпечує доступ до передових алгоритмів. Експериментальний код розроблявся та виконувався в інтерактивному середовищі Jupyter Notebook [88], яке дозволяє поєднувати програмний код, його результати, текстові пояснення та візуалізації в єдиному документі, що є надзвичайно зручним для дослідницької роботи та забезпечує високий рівень відтворюваності експериментів.

Для збору первинних текстових даних з відкритих джерел, зокрема соціальних мереж та новинних порталів, використовувався тандем бібліотек Requests [89] та Beautiful Soup 4 (bs4) [90]. Бібліотека Requests забезпечувала виконання HTTP-запитів та отримання HTML-коду веб-сторінок, тоді як bs4 використовувалася для його парсингу та вилучення необхідного текстового контенту шляхом навігації по DOM-дереву. Такий підхід дозволив створити гнучкий та ефективний інструмент для автоматизованого збору великих масивів неструктурованих даних.

Після збору дані структурувалися та оброблялися за допомогою фундаментальних бібліотек екосистеми Python для науки про дані - Pandas [91] та NumPy[92]. Бібліотека Pandas використовувалася для представлення текстового корпусу та його метаданих у вигляді табличних структур даних (DataFrame), що забезпечує зручні можливості для фільтрації, очистки, агрегації та маніпуляції даними. NumPy слугувала основою для всіх числових обчислень, надаючи ефективні багатовимірні масиви та широкий набір математичних функцій, які є фундаментом для роботи всіх бібліотек машинного навчання.

Процес розмітки даних, що є критично важливим для створення якісної навчальної вибірки, здійснювався за допомогою відкритої платформи Label Studio [93]. Цей вибір був зумовлений кількома ключовими перевагами: гнучкістю у налаштуванні інтерфейсу анотування, підтримкою спільної роботи кількох експертів, а також можливістю інтеграції з моделями машинного навчання для реалізації підходу "людина-в-контурі" (human-in-the-loop). Використання Label Studio дозволило створити стандартизований та

контрольований процес анотування, що забезпечило високу якість та консистентність фінального датасету.

Ядром експериментальної установки є бібліотеки для глибокого навчання. В якості основного фреймворку було обрано PyTorch[86], який на сьогодні є провідним стандартом у дослідницькому середовищі. Його перевагами є динамічний обчислювальний граф ("define-by-run"), що забезпечує гнучкість у побудові складних архітектур, інтуїтивно зрозумілий API, а також потужна підтримка для роботи з GPU.

Для реалізації трансформерних архітектур використовувалася екосистема Hugging Face, зокрема бібліотека Transformers [94]. Ця бібліотека надає доступ до тисяч попередньо навчених моделей (включаючи багатомовні версії BERT та RoBERTa), стандартизованих токенизаторів та високорівневих API для навчання та інференсу (напр., Trainer API). Використання Hugging Face дозволило значно прискорити процес експериментування та забезпечити відповідність реалізації кращим практикам у галузі.

Для візуалізації результатів експериментів, зокрема графіків процесу навчання (зміна функції втрат та метрик якості по епохах), матриць плутанини та розподілу даних, використовувалася бібліотека Matplotlib [95]. Вона надає широкий набір інструментів для створення статичних, анімованих та інтерактивних візуалізацій, що є необхідним для глибокого аналізу та наочного представлення отриманих результатів.

Для реалізації етапу інтерпретації моделі, що є важливою частиною розробленого методу, було залучено бібліотеку SHAP (SHapley Additive exPlanations). Цей інструмент, що базується на теоретико-ігровому підході, дозволяє обчислювати внесок кожного токена у фінальний прогноз моделі, забезпечуючи високий рівень прозорості та пояснюваності її рішень. Використання SHAP дозволило не лише оцінити якість моделі, але й проаналізувати, на які саме лінгвістичні патерни вона спирається при розпізнаванні емоцій.

Однією з ключових цілей дисертаційного дослідження є створення методу, адаптованого до універсальних комп'ютерних систем. Для підтвердження цієї властивості експерименти з оцінки продуктивності (швидкості інференсу, споживання пам'яті) проводилися на трьох гетерогенних апаратних платформах, що представляють різні сегменти обчислювальних систем:

- Малопотужна система: Одноплатний комп'ютер Raspberry Pi 4 з 4 ГБ оперативної пам'яті та 4 ядра ARM Cortex A72. Ця платформа симулює умови роботи на вбудованих системах, смартфонах або пристроях Інтернету речей (IoT) з жорсткими обмеженнями на обчислювальні ресурси та енергоспоживання.
- Сучасна десктопна система: Ноутбук на базі процесора Apple M3 Pro (CPU) 11 процесорних ядер ARM. Ця платформа представляє типове робоче середовище кінцевого користувача або розробника, де обчислення виконуються на потужному, але все ж споживчому процесорі без залучення дискретного GPU.
- Високопродуктивна хмарна система: Сервер на базі процесорів Intel Xeon Scalable та графічного прискорювача NVIDIA A10G з 24 ГБ відеопам'яті (VRAM). Ця конфігурація є типовою для хмарних середовищ та представляє умови для навчання моделей та їх промислової експлуатації у високонавантажених сервісах.

Для наочної демонстрації роботи розробленого методу та моделей було створено програмний прототип у вигляді веб-додатку. Для швидкої розробки інтерактивного інтерфейсу користувача було використано бібліотеку Gradio [85]. Вона дозволяє створювати прості та функціональні веб-інтерфейси безпосередньо з Python-коду, що є ідеальним для прототипування та демонстрації результатів досліджень. Щоб скористатися ПО потрібно спочатку запуснути ПО, потім відкрити браузер та перейти за наступною адресою <http://127.0.0.1:7860/> (або вказаною адресою, яку програма надасть у терміналі після її запуску).

На рис. 4.4 продемонстровано головний інтерфейс розробленого програмного прототипу. Він містить поле для введення тексту ("Введіть текст:"), поле для виведення результату ("Результат розпізнавання:"), кнопки для керування ("Submit", "Clear") та набір готових прикладів для швидкого тестування.

Розпізнавання Емоційного Забарвлення Тексту

Введіть текст для розпізнавання емоційного забарвлення

Введіть текст:

Результат розпізнавання:

Clear Submit Flag

Examples

Я такий щасливий, накінецьто здійснилася моя мрія! Моя мама в лікарні, я дуже хвилююсь за неї.

Рис. 4.4. Загальний вигляд інтерфейсу програмного прототипу

На рис. 4.5 показано результат роботи системи після введення тексту «Я такий щасливий, накінець то здійснилася моя мрія!». Система коректно розпізнала домінуючу емоцію «РАДІСТЬ» та вказала ступінь своєї впевненості у цьому прогнозі (96.51%), що демонструє практичну реалізацію та ефективність розробленого методу.

Розпізнавання Емоційного Забарвлення Тексту

Введіть текст для розпізнавання емоційного забарвлення

Введіть текст:

Я такий щасливий, накінецьто здійснилася моя мрія!

Результат розпізнавання:

РАДІСТЬ (Впевненість: 96.51%)

Clear Submit Flag

Examples

Я такий щасливий, накінецьто здійснилася моя мрія! Моя мама в лікарні, я дуже хвилююсь за неї.

Рис. 4.5. Приклад результату розпізнавання емоційного забарвлення

Таким чином, створена експериментальна установка, що поєднує сучасне програмне забезпечення, відкриті інструменти та гетерогенну апаратну базу, забезпечує всі необхідні умови для проведення всебічного, об'єктивного та відтворюваного дослідження ефективності розробленого методу та моделей. Розроблені програмні застосунки впроваджені в навчальний процес на кафедрі системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" (акт впровадження від 30.04.2026).

4.3 Експериментальне дослідження

Для практичної верифікації розробленого в розділі 3 методу, а також для кількісної оцінки ефективності різних архітектурних рішень в умовах універсальних комп'ютерних систем було сплановано та проведено серію експериментальних досліджень. Метою експерименту є перевірка ключової гіпотези дисертаційної роботи: запропонований модульний метод дозволяє створювати НММ, що забезпечують оптимальний, керований компроміс між точністю розпізнавання емоцій та обчислювальною ефективністю в межах текстів українською мовою. Для досягнення цієї мети було реалізовано зібраний датасет україномовних тестів із соціальних мереж, а також два програмні прототипи модулів розпізнавання, що представляють два полюси спектру "швидкість-точність": архітектуру на основі усереднених векторних представлень (FastText [69]) та архітектуру на основі Трансформера (RoBERTa-base [72]).

Вибір саме цих архітектур як основи для експериментальної реалізації був мотивований необхідністю дослідити фундаментальний компроміс, описаний в алгоритмі вибору архітектури (рис. 3.2). Порівняльний аналіз цих двох моделей дозволяє емпірично оцінити, наскільки великим є приріст у точності при переході до складної архітектури та якою є ціна цього приросту в

одиницях обчислювальних ресурсів при розпізнаванні емоційної тональності в українських текстах:

- FastText реалізує сценарій обмежених ресурсів. Ця модель, що базується на усередненні векторів слів та n -грам символів, представляє полегшене, обчислювально ефективне рішення з мінімальними системними вимогами. Вона слугує базовим рівнем, що демонструє, якої точності можна досягти, практично ігноруючи складний синтаксичний контекст. Її оцінка є критично важливою для визначення доцільності застосування РЕЗТ на пристроях з жорсткими обмеженнями (напр., Raspberry Pi 4).

- RoBERTa-base, на противагу, реалізує сценарій достатніх ресурсів. Ця модель втілює передовий (state-of-the-art) підхід на основі Трансформера з механізмами само-уваги. Завдяки здатності вловлювати складні довготривалі та нелінійні залежності в тексті, очікується, що вона забезпечить значно вищу точність класифікації. Водночас її оцінка дозволить кількісно виміряти обчислювальні витрати, необхідні для досягнення такої точності, що є ключовим для проектування високопродуктивних систем.

Для адаптації цих базових архітектур до специфіки поставленої задачі - багатокласової класифікації однієї домінуючої емоції з-поміж шести базових емоцій (Радість, Смуток, Гнів, Страх, Відраза, Подив) та класу "Нейтральний" - їхні стандартні конфігурації було модифіковано. Поверх вихідних представлень кожної моделі (агрегованого вектора для FastText та вектора [CLS] для RoBERTa-base) було додано уніфікований класифікаційний блок (classification head). Цей блок складається з двох послідовних повнозв'язних шарів з функцією активації ReLU між ними та регуляризаційним шаром Dropout для запобігання перенавчанню [96]. Фінальний вихідний шар містить сім нейронів (за кількістю цільових класів) та використовує функцію активації Softmax для отримання розподілу ймовірностей. Такий вихід гарантує, що сума ймовірностей по всіх класах дорівнює одиниці, що відповідає постановці задачі вибору однієї, найбільш ймовірної емоції. Уніфікація класифікаційної голови дозволяє забезпечити чистоту експерименту, оскільки будь-які

відмінності у результатах будуть зумовлені саме потужністю блоків вилучення ознак, а не різницею у фінальних класифікаторах.

Необхідною передумовою для проведення об'єктивного та відтворюваного експерименту є створення якісного, збалансованого та репрезентативного набору даних. Етап попередньої обробки даних був уніфікованим для обох архітектур, що дозволило забезпечити ідентичні вхідні умови та гарантувати, що відмінності у фінальних результатах будуть зумовлені виключно архітектурними особливостями моделей. Цей процес, що є практичною реалізацією блоку препроцесингу з концептуальної моделі, включав низку послідовних кроків.

Першим кроком було формування первинного «сирого» корпусу, що налічував близько 18 000 україномовних дописів. Дані були випадковим чином відібрані з публічних каналів мережі Telegram та дописів з мережі X (раніше Twitter) у період з жовтня по грудень 2024 року. Вибір цих джерел зумовлений їхньою популярністю в українському медіа-просторі та характерним для них стилем неформальної, емоційно насиченої комунікації. Важливо зазначити, що збір даних проводився без застосування тематичних чи доменних фільтрів. Такий підхід дозволив забезпечити широкий зріз загальноновживаного користувацького дискурсу, а не контенту, штучно прив'язаного до певної тематики (наприклад, лише політичних новин чи відгуків про товари). Це підвищує екологічну валідність корпусу та сприяє кращій узагальнюючій здатності моделей, що на ньому навчаються.

Отриманий сирий корпус містив значну кількість шуму та нерелевантних даних, тому наступним кроком стала його ретельна фільтрація та очистка. Спочатку було проведено фільтрацію за трьома критеріями: мова (залишено лише тексти, ідентифіковані як українські), мінімальна довжина (відкинуто дописи, що містили менше 5 слів, оскільки вони часто є контекстно-неоднозначними) та унікальність (видалено повні дублікати для уникнення витоку даних між навчальною та тестовою вибірками). На етапі очистки з текстів було видалено весь цифровий шум: URL-посилання, імена

користувачів (@mentions), хештеги, HTML-теги та інші спеціальні символи. Також було проведено нормалізацію пунктуації та стандартизацію кодування до формату UTF-8 для забезпечення консистентності даних.

Анотування корпусу здійснювалося за допомогою гібридного підходу "людина-в-контурі" для забезпечення як швидкості, так і високої якості розмітки. На першому етапі було використано велику мовну модель (GPT-4o) [97] для генерації попередніх, кандидатських міток для всього корпусу. На другому, ключовому етапі, кожен текстовий фрагмент з його кандидатською міткою було передано на верифікацію незалежним анотаторам-волонтерам.

Природний розподіл емоцій у текстах є вкрай нерівномірним, що може призвести до упередженості моделі на користь домінуючих класів. Для мінімізації цього ефекту було проведено балансування вибірки. Замість штучного вирівнювання класів за допомогою простої передискретизації або недодискретизації, було застосовано комбінований підхід: спершу було видалено невелику частину прикладів з найбільш представленого "нейтрального" класу, а потім використано техніку зважування класів (class weighting) у функції втрат під час навчання. Цей метод дозволяє зберегти природний розподіл даних, водночас змушуючи модель приділяти більше уваги рідкісним класам.

Фінальним кроком стала токенизація, яка виконувалася окремо для кожної архітектури відповідно до її вимог. Для FastText було застосовано класичну токенизацію на рівні слів з використанням україномовного словника та лематизації. Для RoBERTa-base використовувався її штатний subword-токенізатор на основі SentencePiece, який є оптимальним для роботи зі складною морфологією та невідомими словами.

В результаті цих процедур було сформовано фінальний експериментальний корпус даних. Він складається з 16 000 україномовних текстових фрагментів, розмічених за сімома класами (шість базових емоцій та нейтральний стан). Розподіл прикладів за класами є наступним: смуток (3 144), гнів (2 434), любов (1 463), здивування (1 638), страх (2 161), радість (3 057) та

нейтральний стан (4 103). Цей корпус є збалансованою та достатньо великою основою для надійного навчання, валідації та тестування розроблених моделей. Візуально розподіл класів у корпусі представлено на рис. 4.6.

На основі підготовленого корпусу даних та відповідно до постановки експерименту було реалізовано та навчено дві нейромережеві моделі. Процес побудови та параметризації був специфічним для кожної архітектури, відображаючи її унікальні властивості та вимоги.

Для реалізації сценарію з обмеженими ресурсами було побудовано класифікатор на основі архітектури FastText. Фундаментом моделі є шар вкладень, навчений за допомогою підходу Skip-gram, який прогнозує контекстні слова на основі центрального слова. Цей вибір дозволяє отримати якісні семантичні представлення, що враховують не лише слова, але й їхні підслівні компоненти (символьні n-грами).

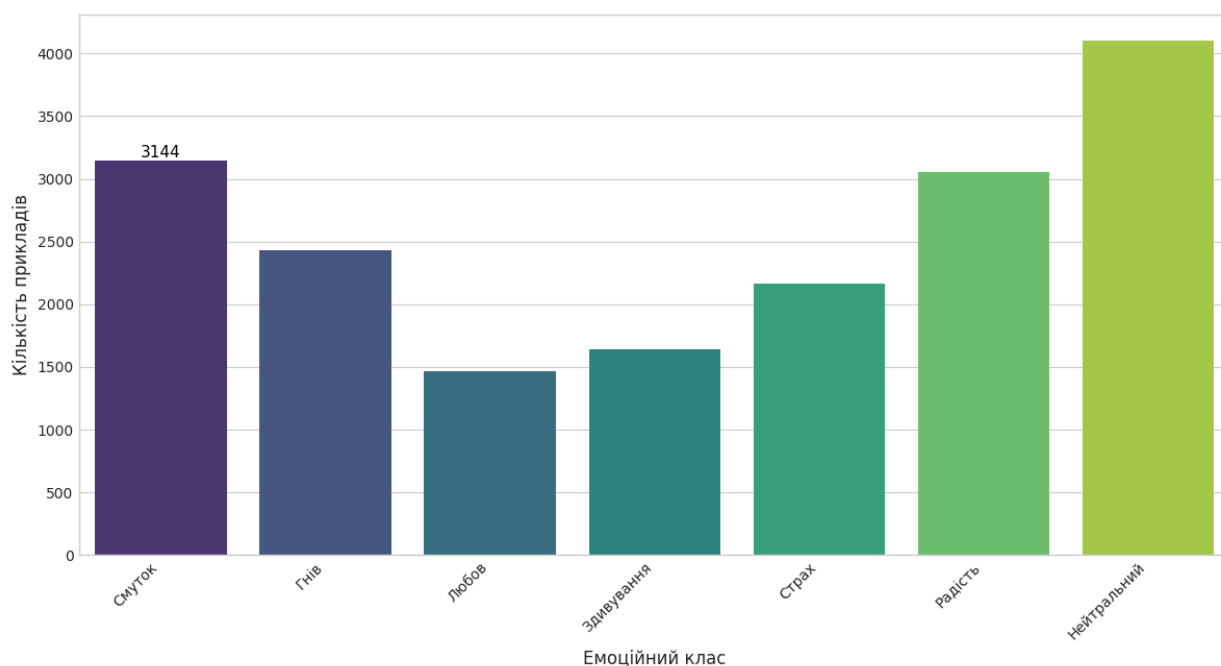


Рис. 4.6. Розподіл емоційних міток в корпусі даних

Параметризація шару вкладень (embeddings) для FastText включала:

- Розмірність вкладень (d): встановлено на рівні 100. Це значення є поширеним компромісом, що дозволяє закодувати достатньо семантичної інформації, не створюючи надто велику модель.

- Розмір контекстного вікна: встановлено на рівні 5 слів (по 5 слів ліворуч та праворуч від центрального), що є стандартним значенням для задач загального призначення.

- Мінімальна частота вживання слова: встановлено на рівні 3. Цей поріг дозволяє відфільтрувати дуже рідкісні слова та можливий "шум" (напр., унікальні друкарські помилки), зберігаючи при цьому майже всю значущу лексику. Такий підхід сприяє навчанню більш надійних векторів, прискорює збіжність та зменшує фінальний розмір моделі без суттєвої шкоди для лексичного покриття.

В результаті було сформовано словник, що налічував приблизно 30 000 унікальних слів. Класифікаційна частина моделі, що реалізує функції F та C , є відносно простою: векторне представлення тексту, отримане шляхом усереднення векторів його слів, подається на класифікаційний блок, що складається з двох повнозв'язних шарів з активацією ReLU та фінального softmax-шару. Загальна кількість параметрів моделі становить приблизно 3 мільйони, що робить її легкою та придатною для швидкого інференсу на CPU.

Для реалізації сценарію з пріоритетом максимальної точності було використано модель на основі архітектури Трансформер. В якості базової моделі було обрано не стандартну багатомовну версію, а спеціалізовану модель Ukrainian-RoBERTa-base [98], яка була додатково попередньо навчена на великому українськомовному корпусі. Цей вибір є критично важливим, оскільки така модель вже адаптована до унікальних лексичних, синтаксичних та семантичних особливостей української мови, що дозволяє досягти значно вищої якості на прикладних задачах.

Параметризація архітектури відповідає стандартній конфігурації RoBERTa-base:

- Кількість шарів кодувальника Трансформера: 12 блоків.

- Кількість голів уваги: 12 у кожному блоці.
- Розмірність прихованого стану (m): 768.

Словник моделі, що базується на subword-токенізації, налічує близько 31 000 токенів і включає як українські, так і англійські (латиницею) підслівні одиниці, що робить його стійким до код-світчингу. Класифікаційний блок, аналогічно до FastText, побудовано поверх виходу [CLS] токена і складається з двох повнозв'язних шарів. Загальна кількість параметрів моделі становить приблизно 125 мільйонів, що на два порядки більше, ніж у FastText, і відображає її значно вищу ємність та обчислювальну складність.

Навчання HMM проводилося з використанням оптимізаційних стратегій, що відповідають їхнім архітектурним особливостям. Для класифікатора FastText було використано класичний оптимізатор - стохастичний градієнтний спуск (SGD) [99]. Розмір міні-пакета (batch size) було встановлено на рівні 32, а початкова швидкість навчання - 0.1 з лінійним спаданням протягом 100 епох навчання (рис 4.7).

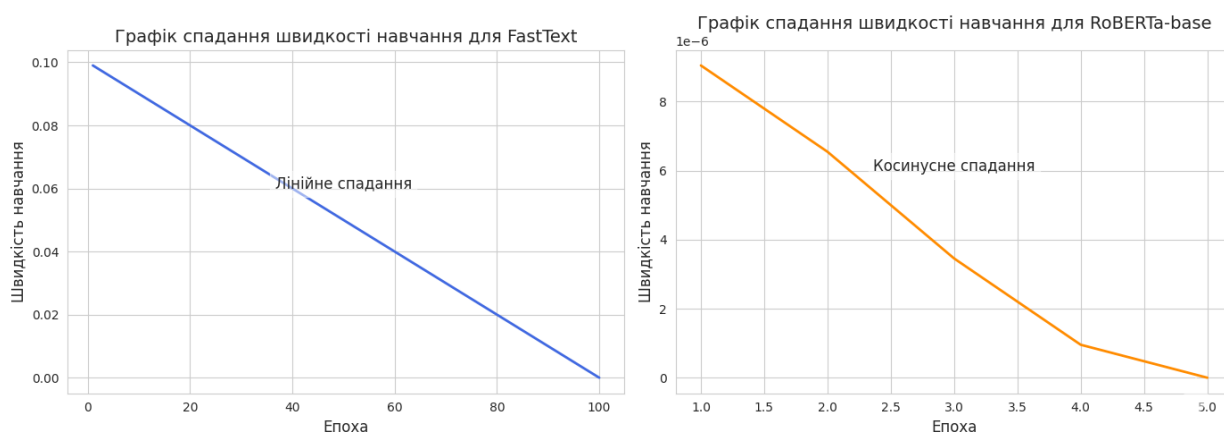


Рис. 4.7. Графіки спадання значень параметра швидкості навчання для експериментальних моделей, на протязі навчання

Для класифікатора RoBERTa-base було застосовано сучасний оптимізатор AdamW [100], який краще підходить для навчання трансформерних моделей. Розмір міні-пакета також становив 32. Початкова швидкість навчання була значно меншою - $1e-5$, що є стандартною практикою

для тонкого налаштування (fine-tuning) попередньо навчених моделей. Для керування швидкістю навчання було використано планувальник з графіком косинусного спадання протягом 5 епох (рис 4.7).

Навчання моделі FastText виконувалося на CPU (Intel Xeon Scalable), тоді як ресурсоемна модель RoBERTa-base навчалася на GPU (NVIDIA A10G з 24 ГБ VRAM). Динаміка процесу навчання, проілюстрована на рисунках 4.7 та 4.8, наочно демонструє відмінності між двома підходами. Функція втрат для FastText (рис. 4.8) поступово знижується з ≈ 1.9 до ≈ 0.35 протягом 100 епох, демонструючи незначні коливання, типові для стохастичної оптимізації на неглибоких архітектурах.

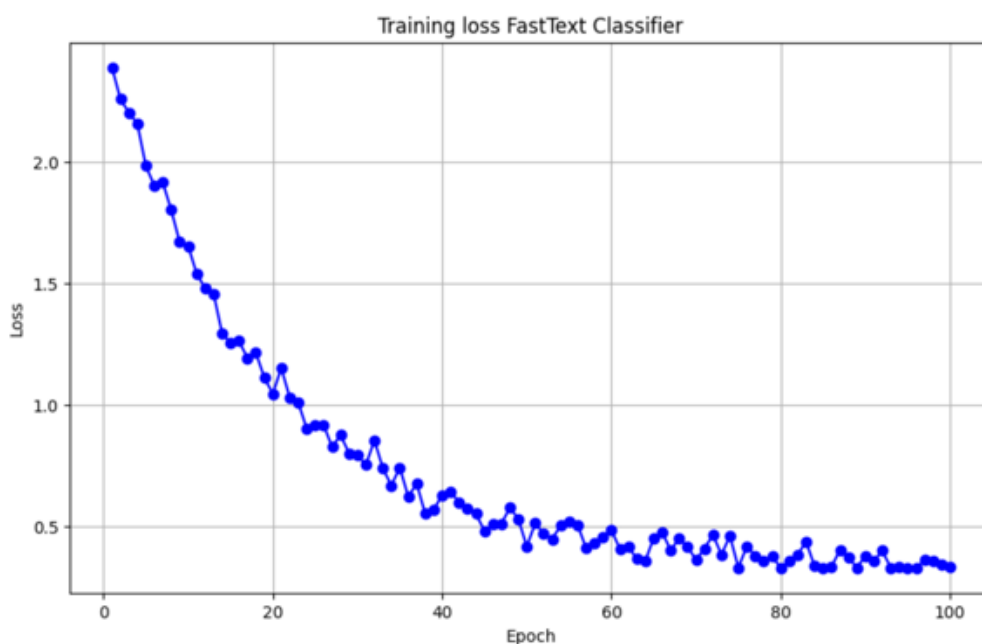


Рис. 4.8. Функція втрат під час навчання для класифікатора FastText

На противагу цьому, крива тонкого налаштування RoBERTa-base (рис. 4.9) показує стрімке падіння втрат з ≈ 1.8 до < 0.1 всього за п'ять епох, після чого вони фактично виходять на плато. У 20 разів вища швидкість збіжності RoBERTa-base підкреслює величезну перевагу використання масштабного попереднього навчання.

Більше того, значно нижчі кінцеві втрати, досягнуті RoBERTa-base, вказують на її вищу здатність вловлювати тонкі семантичні ознаки, які підхід

"мішка слів" у FastText змоделювати не може.



Рис. 4.9. Функція втрат під час навчання для класифікатора RoBERTa-base

Після успішного навчання експериментальних моделей було проведено їх всебічну оцінку та порівняльний аналіз. Метою цього етапу було не лише визначити абсолютні показники ефективності розроблених нейромережових засобів, але й зіставити їх з наявними альтернативними підходами для розв'язання задачі РЕТ в українських текстах. Такий порівняльний аналіз є критично важливим для об'єктивної оцінки переваг та недоліків запропонованого методу в контексті сучасного стану галузі.

Оцінка ефективності розроблених моделей здійснюється відповідно до системи критеріїв, обґрунтованої у другому розділі (рис. 2.4). Вибір альтернативних підходів для порівняння. Зважаючи на обмежену кількість публічно доступних та спеціалізованих класифікаторів для розпізнавання емоцій саме в україномовних текстах, для порівняння було обрано два поширені альтернативні підходи, які часто використовуються на практиці за відсутності спеціалізованих рішень:

Використання пропрієтарної LLM. В якості представника цього класу було обрано модель GPT-4o, яка демонструє передові результати в широкому спектрі завдань обробки природної мови, включно з аналізом тексту багатьма мовами. Цей підхід слугує верхньою межею досяжної якості, показуючи, яких результатів можна досягти, маючи доступ до практично необмежених обчислювальних ресурсів та даних.

Підхід на основі машинного перекладу. Цей метод є поширеним "обхідним шляхом" для низькоресурсних мов. Він полягає у послідовному застосуванні двох моделей: спочатку український текст перекладається англійською мовою (Ukr2Eng Translation [101]), а потім отриманий англійський текст подається на вхід спеціалізованого англійського класифікатора емоцій (Eng-Emotion Classifier [102, 103]). Цей підхід дозволяє оцінити, наскільки ефективною є така стратегія та які втрати якості виникають через потенційне "згладжування" емоційних нюансів під час перекладу.

Критерії та методологія оцінки. Етап валідації, тестування та порівняння всіх чотирьох моделей (FastText, RoBERTa-base, GPT-4o та підхід з перекладом) проводився за комплексним набором критеріїв, що охоплюють як якість, так і ефективність:

- Обчислювальна складність: оцінювалася за кількістю параметрів моделі, що є мірою вимог до пам'яті.
- Швидкість розпізнавання (інференсу): вимірювалася як середній час обробки одного текстового фрагмента довжиною 20 слів на трьох гетерогенних апаратних платформах.
- Якість класифікації: оцінювалася за стандартними метриками Precision, Recall, Accuracy та F1-score, а також за інтегральним показником Quality, розрахованим за формулою (2.19) з рівними коефіцієнтами важливості ($\alpha = \beta = \gamma = \delta = 0.25$).
- Інтерпретованість: аналізувалася за допомогою методу SHAP для розроблених моделей, щоб оцінити прозорість їхніх рішень.

Аналіз результатів якості та обчислювальної складності. Результати порівняльного аналізу якості класифікації та обчислювальної складності представлено в табл. 4.1.

Таблиця 4.1

Порівняльний аналіз моделей розпізнавання емоцій

Назва моделі	Назва метрики					
	Кількість параметрів	Precision	Recall	Accuracy	F1	Quality
FastText	3 М	0,72	0,72	0,72	0,72	0.72
RoBERTa-base	125 М	0,91	0,91	0,91	0,91	0.91
Перекладач + Класифікатор	74М+83М = 157М	0.59	0.59	0.59	0.59	0.59
GPT-4o	—	0.95	0.96	0.96	0.95	0.95

Результати в табл. 4.1 дозволяють зробити кілька ключових висновків. Модель RoBERTa-base, побудована на архітектурі Трансформера, досягла найвищої точності серед усіх моделей, придатних для локального розгортання, отримавши F1-міру 0.91. Це на 19 відсоткових пунктів перевищує результати FastText (F1-міра 0.72), що емпірично підтверджує гіпотезу про те, що глибокі контекстуальні моделі значно ефективніше вловлюють складні емоційні нюанси. Водночас цей приріст точності супроводжується значним збільшенням обчислювальної складності: RoBERTa-base містить 125 мільйонів параметрів, що у 41.7 раза більше, ніж 3 мільйони параметрів FastText.

Підхід, заснований на перекладі, продемонстрував найнижчу продуктивність (F1-міра 0.69). Це підтверджує гіпотезу про суттєву втрату культурно-специфічних та прагматичних емоційних маркерів під час машинного перекладу. Більше того, послідовне застосування двох моделей

робить цей підхід найважчим з точки зору кількості параметрів (157 мільйонів), що робить його неефективним як з точки зору якості, так і з точки зору ресурсів. Незважаючи на те, що GPT-4o досягла найвищої точності класифікації (F1-міра 0.95), її практичне використання в рамках універсальних комп'ютерних систем обмежене низкою фундаментальних недоліків: неможливістю локального розгортання, високою вартістю API-запитів, повною непрозорістю її внутрішньої архітектури ("чорна скринька") та неможливістю контролювати потенційні упередження.

Аналіз швидкості розпізнавання. Результати вимірювання швидкості інференсу на різних апаратних платформах, представлені в табл. 4.2.

Таблиця 4.2

Швидкість інференсу текстового фрагменту з 20 слів

Назва моделі	Апаратна платформа		
	Raspberry Pi4	Apple M3 Pro	NVIDIA A10G
FastText	87 мкс	9 мкс	—
RoBERTa-base	693 ms	50 ms	9 ms
Перекладач + Класифікатор	—	593 мс	73 мс
GPT-4o	2343 мс*		

* апаратна платформа невідома, вказано середній час відповіді на API-запит, що включає мережеві затримки.

Результати табл. 4.2 демонструють значну різницю у швидкості. FastText є абсолютною лідеркою за швидкістю, обробляючи текст за мікросекунди на сучасних CPU. Важливо, що вона є єдиною моделлю, яка показує прийнятний час роботи (менше 1 мс) на малопотужній платформі Raspberry Pi 4. На противагу, RoBERTa-base на CPU працює значно повільніше (50 мс на Apple M3 Pro), але її швидкість кардинально зростає при використанні GPU (9 мс), стаючи придатною для систем реального часу. Підхід з перекладом є

надзвичайно повільним навіть на потужному обладнанні. GPT-4o з часом відповіді понад 2 секунди є абсолютно непридатною для будь-яких інтерактивних застосунків. Ці дані емпірично підтверджують компроміс "точність-швидкість" та обґрунтовують необхідність використання алгоритму вибору архітектури, запропонованого в методі. Аналіз інтерпретованості. На рис. 4.10 продемонстровано пояснення SHAP для прогнозу емоції «Радість» на реченні «Я дуже радий, що ти прийшов на мій день народження», отримане для моделі RoBERTa-base. Починаючи з базового значення на рівні всього корпусу (base value = 0.448), метод SHAP адитивно додає внесок кожного токена. Емоційно забарвлені слова, такі як «радий» та інтенсифікатор «дуже» (показані червоним), роблять найбільший позитивний внесок, сукупно підвищуючи логарифм шансів до фінального значення 0.993. Службові слова, як-от «ти», «мій», та прийменники (показані синім), створюють лише незначні негативні зміщення, що є очікуваною та логічною поведінкою. Таким чином, підтверджено, що рішення класифікатора ґрунтується на семантично релевантних, контентних словах, що є носіями емоційної валентності. Це надає інтуїтивно зрозумілу декомпозицію того, як окремі токени формують фінальну емоційну мітку, що підвищує довіру до результатів.



Рис. 4.10. Візуалізація декомпозиції прогнозу «Радість» класифікатора RoBERTa-base на внески окремих tokenів за допомогою SHAP

Отже, отримані результати повністю підтверджують ефективність запропонованих нейромережевих модулів. Вони наочно ілюструють існування компромісу між якістю класифікації та обчислювальними вимогами. Розроблений модульний підхід дозволяє ефективно керувати цим компромісом, обираючи оптимальну конфігурацію моделі відповідно до

конкретних практичних вимог щодо точності, швидкості та наявних обчислювальних ресурсів. На завершення експериментальних досліджень проведено експертне оцінювання ефективності розробленої системи розпізнавання за відповідністю до характеристик, визначених у п.1.2 дисертаційної роботи. Отримані результати наведено в табл. 4.3.

Таблиця 4.3

Оцінка характеристик розробленої системи розпізнавання

№	Опис характеристик	Оцінка
N ₁	Розпізнавання основних емоцій	1
N ₂	Розпізнавання лише сентиментів	0
N ₃	Розпізнавання на рівні окремих висловів	1
N ₄	Розпізнавання на рівні речень	1
N ₅	Розпізнавання на рівні документу	1
N ₆	Використання лексичного аналізу	0
N ₇	Використання сучасних нейромережових засобів	1
N ₈	Використання відомих наборів даних	0
N ₉	Використання власних наборів даних	1
N ₁₀	Можливість врахування специфіки української мови	1
N ₁₁	Використання комбінації моделей для розпізнавання	1
N ₁₂	Використання однієї моделі для розпізнавання	0
N ₁₃	Точність розпізнавання	1
N ₁₄	Оперативність розпізнавання	1

Результат порівняння даних наведених в табл. 4.3 з даними табл. 1.2, свідчить, що розроблена система розпізнавання емоційної тональності фрагментів тексту перевершує найбільш відомі системи аналогічного призначення з точки зору можливості комплексного забезпечення точності і оперативності розпізнавання (якості розпізнавання), що підтверджує досягнення мети дисертаційних досліджень.

4.4. Висновки до четвертого розділу

Даний розділ присвячено вирішенню науково-практичної задачі розробки та дослідження системи розпізнавання емоційної тональності текстових фрагментів в універсальних комп'ютерних системах. Основні результати розділу наступні:

- На базі запропонованих моделей та методу розроблено архітектуру програмної системи розпізнавання емоційної тональності фрагментів тексту, що дозволяє підвищити якість розпізнавання, адаптована для інтеграції в універсальні комп'ютерні системи моніторингу інформаційного простору, а також може бути використана для створення спеціалізованих інструментальних засобів відповідного призначення.

- Створено анотований україномовний корпус емоційно забарвлених текстів, що може бути використаний науковою спільнотою для подальших досліджень у галузі обробки природної мови.

- В результаті проведених досліджень з використанням україномовних текстів визначено, що запропоновані рішення дозволяють приблизно на 26% підвищити точність розпізнавання емоційної тональності при розгортанні засобів розпізнавання на високопродуктивних системах або приблизно на порядок підвищити оперативність розпізнавання, що забезпечує можливість розгортання таких засобів на мобільних пристроях та свідчить про підвищення ефективності процесу розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах за рахунок підвищення якості розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

- Розроблені програмні застосунки, що реалізують запропонований метод, впроваджені в навчальний процес на кафедрі системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" (акт впровадження від 30.04.2026).

ВИСНОВКИ

В дисертаційній роботі вирішено актуальне наукове завдання дослідження, яке полягає у розробці нейромережових моделей та методу, що забезпечують підвищення ефективності розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах в умовах обмежених обчислювальних ресурсів з урахуванням лінгвістичної специфіки української мови.

Проведені дослідження дозволяють зробити наступні висновки:

1. Визначено, що важливим напрямком підвищення ефективності розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах є вдосконалення нейромережових засобів розпізнавання. Для цього необхідно розвинути методологічну базу, розробити метод розпізнавання емоційної тональності та анотований корпус україномовних текстів, які забезпечать їх адаптацію до застосування в умовах обмежених обчислювальних ресурсів з урахуванням лінгвістичної специфіки української мови, а також розробити та дослідити систему емоційної тональності фрагментів тексту, яка ґрунтується на запропонованих моделях і методах.

2. Отримала подальший розвиток методологічна база нейромережевого розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах, що за рахунок розроблених моделей та системи критеріїв оцінювання ефективності забезпечує можливість створення ефективних нейромережових методів розпізнавання.

3. Розроблено метод розпізнавання емоційної тональності текстових фрагментів з використанням нейромережових засобів в універсальних комп'ютерних системах що за рахунок комплексного використання моделі побудови нейромережових засобів, удосконаленої системи критеріїв

оцінювання та розробленого анотованого корпусу україномовних текстів, забезпечує можливість адаптації архітектури засобу розпізнавання до умов застосування та розширення лексичного покриття предметної області.

4. Розроблено та досліджено систему емоційної тональності фрагментів тексту, яка ґрунтується на запропонованих моделях і методах та адаптована до інтеграції в універсальні комп'ютерні системи моніторингу інформаційного простору, а також може бути використана для створення спеціалізованих інструментальних засобів відповідного призначення. В результаті проведених досліджень з використанням україномовних текстів визначено, що запропоновані рішення дозволяють приблизно на 26% підвищити точність розпізнавання емоційної тональності при розгортанні засобів розпізнавання на високопродуктивних системах або приблизно на порядок підвищити оперативність розпізнавання, що забезпечує можливість розгортання таких засобів на мобільних пристроях та свідчить про підвищення ефективності процесу розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах за рахунок підвищення якості розпізнавання з урахуванням обмежень щодо обчислювальної ресурсоемності, а також лінгвістичних особливостей української мови.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Sharma R. Sentiment Analysis Market Research Report 2033. *Market Intelo*. Last updated: August 25, 2025. URL: <https://marketintel.com/report/sentiment-analysis-market> (дата звернення: 02.10.2025).
2. Joshi, P., Santy, S., Budhiraja, A., Bali, K., & Choudhury, M. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics*. 2020. pp. 6282-6293
3. Strubell, E., Ganesh, A., & McCallum, A. Energy and Policy Considerations for Deep Learning in NLP. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics*, 2019 pp. 3645-3650.
4. Дичка, І. А., Терейковський, І. А., Коровій, О. С., Терейковська, Л. О., & Романкевич, В. О. Оцінка ефективності засобів розпізнавання емоційної тональності фрагментів тексту. *Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки*, 34(3). 2023. 114–119. <https://doi.org/10.32782/2663-5941/2023.3.1/20>
5. Коровій, О. С., & Терейковський, І. А.. Концептуальна модель процесу визначення емоційної тональності тексту. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, 55. 2024. 115–123. <https://doi.org/10.36910/6775-2524-0560-2024-55-14>
6. Терейковський, І. А., & Коровій, О. С. Обґрунтування актуальності досліджень в області розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах. В *Прикладна математика та комп'ютинг (ПМК 2022)* (м. Київ, НТУУ «КПІ ім. Ігоря Сікорського», 20–22 листопада 2022 р.). – С. 361–364.
7. Коровій, О. С., & Терейковський, І. А.. Метод побудови нейромережових засобів розпізнавання емоційного забарвлення текстів в універсальних комп'ютерних засобах. *Комп'ютерно-інтегровані*

технології: освіта, наука, виробництво, 60. 2025 – С. 182–189.
<https://doi.org/10.36910/6775-2524-0560-2025-60-19>

8. Korovii, O., & Petrashenko, A. Adaptation of distilling knowledge method in natural language processing for sentiment analysis. In Z. Hu, S. Petoukhov, F. Yanovsky, & M. He (Eds.), *Advances in computer science for engineering and manufacturing (ISEM 2021)* (Lecture Notes in Networks and Systems, Vol. 463). Springer, Cham. 2022. pp 170–180.
https://doi.org/10.1007/978-3-031-03877-8_15

9. Tereikovskiy, I., & Korovii, O. Conceptual model of the process for determining the emotional coloring and tonality of text fragments. In *Applied Mathematics and Computing* (AMC 2024) (м. Київ, НТУУ «КПІ ім. Ігоря Сікорського», 20–22 листопада 2022 р.).

10. Зайцев, В. Г., & Коровій, О. С.. Проблематика розпізнавання емоційної тональності фрагментів тексту в універсальних комп'ютерних системах. В “IX Міжнародна науково-технічна Internet-конференція “Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами”. 2022. НУХТ.

11. Терейковський, І. А., Коровій, О. С., Дичка, І. А., Романкевич, В. О., & Тарасенко-Клятченко, О. В. Text analysis (Свідоцтво про реєстрацію авторського права на твір № 7087). 2024. Національний орган інтелектуальної власності.

12. Терейковський, І. А., Коровій, О. С., Дичка, І. А., Романкевич, В. О., & Тарасенко-Клятченко, О. В. *Emotion Recognition* (Свідоцтво про реєстрацію авторського права на твір № 7433). 2025. Національний орган інтелектуальної власності. <https://sis.nipo.gov.ua/uk/search/detail/1854403/>

13. Bagitova, K., Tereikovskiy, I., Babayev, I., Tereikovska, L., & Tereikovskiy, O. (2023). Model for processing images of online social networks used to recognize political extremism. *Journal of Mathematics, Mechanics and*

<https://doi.org/10.26577/JMMCS2023v119i3a8>

14. Iosifov, I., Iosifova, O., & Sokolov, V. Natural language technology to ensure the safety of speech information. *In Workshop on Cybersecurity Providing in Information and Telecommunication Systems II*. 2022. p. 216–226.

15. Ronen Feldman. Techniques and applications for sentiment analysis. *Commun. ACM* 56, 4. 2013. p. 82–89. <https://doi.org/10.1145/2436256.2436274>

16. Mussiraliyeva, S., Bagitova, K., & Sultan, D. Social media mining to detect online violent extremism using machine learning techniques. *International Journal of Advanced Computer Science and Applications*, 14(6). 2023. <http://dx.doi.org/10.14569/IJACSA.2023.01406146>

17. Tereikovskiy, I. A., & Korovii, O. S. A model for constructing neural network systems for recognizing emotions of text fragments. *Herald of Advanced Information Technology*, 8(2). 2025. p. 197–208. <https://doi.org/10.15276/hait.08.2025.12>

18. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics. 2019. pp. 4171–4186.

19. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. “Roberta: A robustly optimized bert pretraining approach”. *arXiv*, 2019. DOI: <https://doi.org/10.48550/arXiv.1907.11692>

20. Radford, A., Narasimhan, K., Salimans, T. and Sutskever, I. Improving Language Understanding by Generative Pre-Training. *OpenAI*. 2018

21. Іосіфов, Є. А. Комплексний метод по автоматичному розпізнаванню природньої мови та емоційного стану. *Кібербезпека: освіта, наука, техніка*, 3(19). 2023. – С. 146–164. <https://doi.org/10.28925/2663-4023.2023.19.146164>

22. Іосіфов, Є. А., Соколов, В. Ю. Методи аналізу природної мови та застосування нейронних мереж в кібербезпеці. *Кібербезпека: освіта, наука, техніка*, 4(24), 2024а. – С. 398–414. <https://doi.org/10.28925/2663-4023.2024.24.398414>

23. Іосіфов, Є. А., Соколов, В. Ю. Порівняльний аналіз методів, технологій, сервісів та платформ для розпізнавання голосової інформації в системах забезпечення інформаційної безпеки. *Кібербезпека: освіта, наука, техніка*, 1(25). 2024б. – С. 468–486.

24. Корченко, О., Терейковський, І., Дичка, І., Романкевич, В., & Терейковська, Л. Виявлення маніпулятивної складової у текстових повідомленнях засобів масової інформації у контексті захисту вітчизняного кіберпростору. *Кібербезпека: освіта, наука, техніка*, 1(29). 2025. – С. 27–40. <https://doi.org/10.28925/2663-4023.2025.29.839>

25. Марценюк, М., Козачок, В., Богданов, О., Іосіфов, Є., & Брже夫ська, З. Аналіз методів виявлення дезінформації в соціальних мережах за допомогою машинного навчання. *Кібербезпека: освіта, наука, техніка*, 2(22). 2023. – С. 148–155. <https://doi.org/10.28925/2663-4023.2023.22.148155>

26. Horák, A., Baisa, V., & Herman, O. Technological approaches to detecting online disinformation and manipulation. In M. Gregor & P. Mlejnková (Eds.), *Challenging online propaganda and disinformation in the 21st century. Political Campaigning and Communication. Palgrave Macmillan, Charm*. 2021. p. 139–166. https://doi.org/10.1007/978-3-030-58624-9_5

27. Oberländer, L. A. M., & Klinger, R. Token sequence labeling vs. clause classification for English emotion stimulus detection. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics. Association for Computational Linguistics*. 2020. pp. 58–70. <https://aclanthology.org/2020.starsem-1.7/>

28. Moores, B., Mago, V.K. A Survey on Automated Sarcasm Detection on Twitter. *ArXiv*. 2022. DOI: <https://doi.org/abs/2202.02516>

29. Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics*. 2020. p. 4067–4076. <https://doi.org/10.18653/v1/2020.acl-main.374>
30. Mickel Hoang, Oskar Alija Bihorac, and Jacobo Rouces. Aspect-Based Sentiment Analysis using BERT. *Proceedings of the 22nd Nordic Conference on Computational Linguistics. Turku, Finland. Linköping University Electronic Press*. 2019. P. 187-191. <https://aclanthology.org/W19-6120.pdf>
31. Wenxuan Shi, Fei Li, Jingye Li, Hao Fei, and Donghong Ji. (). Effective Token Graph Modeling using a Novel Labeling Strategy for Structured Sentiment Analysis. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Dublin, Ireland. Association for Computational Linguistics*. 2022. p.4232–4241. <https://doi.org/10.18653/v1/2022.acl-long.291>
32. A. S. Imran, S. M. Daudpota, Z. Kastrati and R. Batra. Cross-Cultural Polarity and Emotion Detection Using Sentiment Analysis and Deep Learning on COVID-19 Related Tweets. *IEEE Access*. vol. 8. 2020. p. 181074-181090. <https://doi.org/10.1109/ACCESS.2020.3027350>
33. Hui Wu and Xiaodong Shi. Adversarial Soft Prompt Tuning for Cross-Domain Sentiment Analysis. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Dublin, Ireland. Association for Computational Linguistics*. 2022. p. 2438-2447. <https://doi.org/10.18653/v1/2022.acl-long.174>
34. Терейковська Л. О. Методологія автоматизованого розпізнавання емоційного стану слухачів системи дистанційного навчання: дис. докт. техн. наук: 05.13.06 – Інформаційні Технології. Київ. 2023. 395 с.
35. Терейковський, І. А., Бушуєв, Д. А., & Терейковська, Л. О. Штучні нейронні мережі: базові положення [Навчальний посібник]. КІІІ ім. Ігоря Сікорського. 2022

36. Mussiraliyeva, S., Bolatbek, M., Omarov, B., & Bagitova, K. Detection of extremist ideation on social media using machine learning techniques. *In Advances in Intelligent Systems and Computing. Springer.* 2020. p. 535–544. https://doi.org/10.1007/978-3-030-63007-2_58
37. Livingstone, S. R., & Russo, F. A. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PLOS ONE*, 13(5), e0196391. 2018. <https://doi.org/10.1371/journal.pone.0196391>
38. Shirani, A., & Nilchi, A. R. N. Speech emotion recognition based on SVM as both feature selector and classifier. *International Journal of Image, Graphics and Signal Processing*, 8(4), 2016. p. 39–45. <https://doi.org/10.5815/ijigsp.2016.04.05>
39. Shirani, A., & Nilchi, A. R. N. Speech emotion recognition based on SVM as both feature selector and classifier. *International Journal of Image, Graphics and Signal Processing*, 8(4), 2016 p. 39-45. <https://doi.org/10.5815/IJIGSP.2016.04.05>
40. Alonso, M.A.; Vilares, D.; Gómez-Rodríguez, C.; Vilares, J. Sentiment Analysis for Fake News Detection. *Electronics* 2021, 10(11), p. 1348. <https://doi.org/10.3390/electronics10111348>
41. Hu, Z., Tereykovskiy, I., Tereykovska, L., & Pogorelov, V. (). Determination of structural parameters of multilayer perceptron designed to estimate parameters of technical systems. *Intelligent Systems and Applications*, 10, 2017. p. 57–62.
42. Iosifova, O., Iosifov, I., & Sokolov, V. Analysis of automatic speech recognition methods. *In Workshop on Cybersecurity Providing in Information and Telecommunication Systems.* 2021. pp. 252–257.
43. Kuchaiev, O., Li, J., Nguyen, H., Hrinchuk, O., Leary, R., Ginsburg, B., ... & Cohen, J. NeMo: A toolkit for building AI applications using neural modules. *arXiv*. 2019. URL: <https://arxiv.org/abs/1909.09577>

44. Romanovskyi, O., et al. Automated pipeline for training dataset creation from unlabeled audios for automatic speech recognition. *In Advances in Computer Science for Engineering and Education IV*. Springer. 2021. pp. 25–36 https://doi.org/10.1007/978-3-030-80472-5_3

45. Терейковський, І. А., & Корченко, А. О. . Інтелектуалізовані методи захисту інформації: нейронні мережі в захисті інформації [Навчальний посібник]. *KPI ім. Ігоря Сікорського*. 2022 URL: <https://ela.kpi.ua/server/api/core/bitstreams/80378d5e-8b7e-4dd7-b8a6-9d504b4a8845/content> (дата звернення 15.10.2025)

46. Toliupa, S., Kulakov, Y., Tereikovskiy, I., Tereikovskiy, O., Tereikovska, L., & Nakonechnyi, V. Keyboard dynamic analysis by AlexNet type neural network. *In 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET) 2020*. pp. 416–420. <https://doi.org/10.1109/TCSET49122.2020.235466>

47. Tereikovskiy, I., Parkhomenko, I., Toliupa, S., & Tereikovska, L. Markov model of normal conduct template of computer systems network objects. *In 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET 2018)*. IEEE. 2018. pp. 498–501. <https://ieeexplore.ieee.org/document/8336143>

48. Opolska, A., Głowacka, D., & Pędziewiatr, M. Irony and Sarcasm Detection for Polish Language using a Lexicon-based Approach. *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop*. Association for Computational Linguistics. 2021. pp. 160–168.

49. Макогон, В. В., & Самохін, І. В. Методи та засоби сентимент-аналізу текстів українською мовою. *Системи управління, навігації та зв'язку*, 2(50), 2020. – С. 96-100.

50. Gilardi, Fabrizio & Alizadeh, Meysam & Kubli, Maël. ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks. *Proc. Natl. Acad. Sci. U.S.A.* 120 (30) e2305016120, 2023. <https://doi.org/10.1073/pnas.2305016120>

51. Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. Green AI. *Communications of the ACM*, 63(12), 2020. p. 54–63.
52. Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? *In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. Association for Computing Machinery*. 2021. pp. 610–623.
53. Wang, X., Kim, H., Rahman, S., Mitra, K., & Miao, Z. Human-LLM collaborative annotation through effective verification of LLM labels. *In Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, May. p. 1–21.
54. Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint*. <https://doi.org/10.48550/arXiv.1412.6980>
55. I. Tereikovskiy, Z. Hu, D. Chernyshev, L. Tereikovska, O. Korystin, and O. Tereikovskiy. The Method of Semantic Image Segmentation Using Neural Networks. *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, vol. 14, no. 6, 2022, pp. 1-14. <https://doi.org/10.5815/ijigsp.2022.06.01>
56. S. Toliupa, Y. Kulakov, I. Tereikovskiy, O. Tereikovskiy, L. Tereikovska, and V. Nakonechnyi. Keyboard Dynamic Analysis by Alexnet Type Neural Network. *In 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, 2020.* pp. 416-420. <https://doi.org/10.1109/TCSET49122.2020.235466>
57. Lundberg, S. M., & Lee, S. I. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017. P. 4765–4775.
58. Ribeiro, M. T., Singh, S., & Guestrin, C. (). "Why should I trust you?": Explaining the predictions of any classifier. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016. pp. 1135–1144.

59. Tan, K. L. Lee, C. P. Anbananthen, K. S. M., Lim, K. M. RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent Neural Network. *In IEEE Access*. Vol. 10, 2022. p. 21517–21525. <https://doi.org/10.1109/ACCESS.2022.3152828>.
60. Tariq Abdullah & Ahmed Ahmet. Deep Learning in Sentiment Analysis: Recent Architectures. *ACM Comput. Surv.* 55, 8, Article 159, 2022. pp. 1-37. <https://doi.org/10.1145/3548772>
61. Korchenko, O., Tereikovskiy, I., Ziubina, R., Tereikovska, L., Korystin, O., Tereikovskiy, O. & Karpinskyi, V. Modular Neural Network model for biometric authentication of personnel in critical infrastructure facilities based on facial images. *Applied Sciences*. 2025. p. 2553. <https://doi.org/10.3390/app15052553>
62. Tereikovska, L. & Tereikovskiy, I. Conceptual principles of neuronal recognition of phonemes in the voice signal of distance learning system members. *Science, Technology and Innovation in the Modern World: Scientific Monograph*. Baltija Publishing. 2023. <https://doi.org/10.30525/978-9934-26-364-4-5>
63. Ben-David, S., Blitzer, J., Crammer, K., & Pereira, F. A theory of learning from different domains. *Machine learning*, 79(1), 2010. P. 151-175.
64. Pang, J., Wei, J., Shah, A. P., Zhu, Z., Wang, Y., Qian, C., ... & Wei, W. Improving Data Efficiency via Curating LLM-Driven Rating Systems. *arXiv preprint*. 2024. <https://doi.org/10.48550/arXiv.2410.10877>
65. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16. 2002. p. 321-357.
66. Bostrom, K., & Durrett, G. Byte Pair encoding is Suboptimal for Language Model Pretraining. *In Findings of the Association for Computational Linguistics: EMNLP*. 2020. – p. 4617-4624. <https://doi.org/10.18653/v1/2020.findings-emnlp.414>

67. Kudo, T., & Richardson, J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 2018. pp. 66-71. <https://doi.org/10.18653/v1/D18-2012>
68. A. Petrescu, C. O. Truica, E. S. Apostol and A. Paschke. EDSA-Ensemble: an Event Detection Sentiment Analysis Ensemble Architecture. *in IEEE Transactions on Affective Computing, Volume: 16, Issue: 2*. 2024. P. 555 – 572. <https://doi.org/10.1109/TAFFC.2024.3434355>
69. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T. Bag of tricks for efficient text classification. *15th Conference of the European Chapter of the Association for Computational Linguistics Eacl. Proceedings of Conference*. 2017, pp. 427–431. <https://doi.org/10.18653/v1/e17-2068>
70. Mikolov, T., Chen, K., Corrado, G., & Dean, J. . Efficient estimation of word representations in vector space. arXiv preprint 2013arXiv:1301.3781.
71. Pennington, J., Socher, R., & Manning, C. D. GloVe: Global vectors for word representation. *In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics*. 2014. p. 1532–1543. <https://doi.org/10.3115/v1/D14-1162>
72. Sazan, S. A., Miraz, M. H., & Rahman, A. B. M. M. Enhancing depressive post detection in Bangla: A comparative study of TF-IDF, BERT and FastText embeddings. *Annals of Emerging Technologies in Computing*, 8(3), 2024, p. 34–49. <https://doi.org/10.33166/AETiC.2024.03.003>
73. Hochreiter, Sepp & Schmidhuber, Jürgen. Long Short-Term Memory. *Neural Computation*, 9(8), 1997, p. 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
74. Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. Attention is all you need. *Advances in*

Neural Information Processing Systems. 31st Conference on Neural Information Processing Systems (NIPS 2017), 2017, pp. 5999 – 6009

75. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv*. 2019. <https://doi.org/10.48550/arXiv.1907.11692>

76. Goodfellow, I., Bengio, Y., & Courville, A. Deep Learning. Cambridge, MA: MIT Press. 2016. (Available online at <http://www.deeplearningbook.org>)

77. Xie, P., Gu, H., & Zhou, D. Modeling sentiment analysis for educational texts by combining BERT and FastText. *In Proceedings of the 6th International Conference on Computer Science and Technologies in Education (CSTE 2024)*, 2024. pp. 195–199.

78. Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. GoEmotions: A dataset of fine-grained emotions. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). Association for Computational Linguistics*. 2020. pp. 4040–4054 <https://doi.org/10.18653/v1/2020.acl-main.372>

79. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 2017. p. 135–146. https://doi.org/10.1162/tacl_a_00051

80. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. Distributed representations of words and phrases and their compositionality. *In Advances in Neural Information Processing Systems* 26. 2013. pp. 3111–3119.

81. Song, X., Salcianu, A., Song, Y., Dopson, D., & Zhou, D. Fast wordpiece tokenization. *In Proceedings of the 2021 conference on empirical methods in natural language processing*. 2021. pp. 2089-2103. <https://doi.org/10.18653/v1/2021.emnlp-main.160>

82. Kudo, T. Subword regularization: Improving neural network translation models with multiple subword candidates. *In Proceedings of the 56th*

Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018. pp. 66-75. <https://doi.org/10.18653/v1/P18-1007>

83. Dahouda, M. K., & Joe, I. A deep-learned embedding technique for categorical features encoding. *IEEE access*, 9. 2021. P. 114381-114391. <https://doi.org/10.1109/ACCESS.2021.3104357>

84. Kumar, J. A., Balakrishnan, M., & Wan Yahaya, W. A. J. (). Emotional design in multimedia learning: How emotional intelligence moderates learning outcomes. *International Journal of Modern Education and Computer Science*, 8(5), 2016, P. 54-63. <https://doi.org/10.5815/ijmecs.2016.05.07>

85. Abid, A., Abdalla, A., & Das, D. Gradio: Hassle-free sharing and testing of ML models in the wild. *arXiv preprint*. 2019. <https://doi.org/10.48550/arXiv.1906.02569>

86. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Chintala, S. PyTorch: An imperative style, high-performance deep learning library. *In Advances in Neural Information Processing Systems* 32. 2019. pp. 8026-8037. <https://dl.acm.org/doi/10.5555/3454287.3455008>

87. Python Software Foundation. Python Language Reference, version 3.11. 2023. URL: <https://www.python.org> (дата звернення: 28.10.2025)

88. Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., ... & Ivanov, P. Jupyter Notebooks – a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. 2016, pp. 87-90. IOS Press.

89. Reitz, K. Requests: HTTP for Humans. 2019. URL: <https://requests.readthedocs.io> (дата звернення: 28.10.2025)

90. Richardson, L. Beautiful Soup Documentation. 2023 URL: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/> (дата звернення: 28.10.2025)

91. McKinney, W. Data structures for statistical computing in python. *In Proceedings of the 9th Python in Science Conference Vol. 445*, 2010. pp. 51-56.

92. Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. Array programming with NumPy. *Nature*, 585 (7825), 2020. P. 357-362.
93. Tkachenko, M., et al. (2020). Label Studio: A versatile data annotation tool. URL: <https://labelstud.io> (дата звернення: 28.10.2025)
94. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020. pp. 38-45.
95. Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 2007, p. 90-95.
96. Mankolli, E., & Guliashki, V. Machine learning and natural language processing: Review of models and optimization problems. In *International Conference on ICT Innovations*. Cham: Springer International Publishing. 2020 pp. 71-86
97. OpenAI. GPT-4o system card. URL: <https://openai.com/index/gpt-4o-system-card/> (дата звернення: 29.10.2025)
98. V. Kurnosov. Ukrainian-RoBERTa-base. *Huggingface*. 2023. URL: <https://huggingface.co/ukr-models/xlm-roberta-base-uk> (дата звернення: 29.10.2025)
99. Bottou, L. Stochastic gradient descent tricks. In *Neural networks: tricks of the trade: second edition*, 2012. pp. 421-436. <https://doi.org/10.1007/978-3-642-35289-8>
100. Loshchilov, I., & Hutter, F. Decoupled weight decay regularization. *arXiv. Preprint*. 2017. <https://doi.org/10.48550/arXiv.1711.05101>
101. Mickus, T., Grönroos, S. A., Attieh, J., Boggia, M., De Gibert, O., Ji, S., ... & Tiedemann, J. MAMMOTH: Massively multilingual modular open translation Helsinki. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 2024, March. pp. 127-136.

102. J. Hartmann. Emotion English DistilRoBERTa-base. 2022.
URL: <https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/>
(дата звернення: 29.10.2025)
103. Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., & Mihalcea, R. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *In Proceedings of the 57th annual meeting of the association for computational linguistics*. 2019. pp. 527-536.

ДОДАТОК А. ПРОГРАМНИЙ КОД

Web_interface.py

```

import torch
import gradio as gr
from transformers import AutoTokenizer, AutoModelForSequenceClassification,
AutoModelForSeq2SeqLM, pipeline

device = 'mps'
# Step 1: Load the pretrained emotion recognition pipeline via Hugging Face
translation_model = AutoModelForSeq2SeqLM.from_pretrained("Helsinki-NLP/opus-
mt-uk-en")
translation_tokenizer = AutoTokenizer.from_pretrained("Helsinki-NLP/opus-mt-
uk-en")

translation_pipe = pipeline('translation', model=translation_model,
tokenizer=translation_tokenizer, device=device)

emotion_model = AutoModelForSequenceClassification.from_pretrained("j-
hartmann/emotion-english-distilroberta-base")
emotion_tokenizer = AutoTokenizer.from_pretrained("j-hartmann/emotion-
english-distilroberta-base")

emotion_pipe = pipeline('text-classification', model=emotion_model,
tokenizer=emotion_tokenizer, device=device)

# Step 2: Define the function for emotion recognition
def predict_emotion(text: str) -> str:
    # Predict emotion probabilities
    english_text = translation_pipe(text,
max_length=512)[0]['translation_text']

    predictions = emotion_pipe(english_text, top_k=7)
    # predictions is a list of dictionaries with 'label' and 'score'
    # e.g. [{'label': 'joy', 'score': 0.78}, ... ]

    # Find the emotion label with the highest score
    best_emotion = max(predictions, key=lambda x: x["score"])
    emotion_label = best_emotion["label"]
    confidence_score = best_emotion["score"]

```

```

eng_emotion_to_ukr = {
    "joy": "радість",
    "anger": "гнів",
    "disgust": "відраза",
    "sadness": "сум",
    "fear": "страх",
    "surprise": "здивування",
    "neutral": "нейтральність"
}

# Return a formatted string with the top predicted emotion and confidence
return f"{eng_emotion_to_ukr[emotion_label].upper()} (Впевненість:
{confidence_score*100:.2f}%) "

# Step 3: Build the Gradio interface
interface = gr.Interface(
    fn=predict_emotion,
    inputs=gr.Textbox(lines=2, label="Введіть текст:"),
    outputs=gr.Text(lines=2, label="Результат розпізнавання:"),
    title="Розпізнавання Емоційного Забарвлення Тексту",
    description="Введіть текст для розпізнавання емоційного забарвлення",
    examples=['Я такий щасливий, накінецьто здійснилася моя мрія!', 'Моя мама
в лікарні, я дуже хвилююсь за неї.']
)

# Step 4: Launch the interface
if __name__ == "__main__":
    interface.launch()

```

train_fasttext.py

```

from collections import defaultdict
import string
import json

# unsupervised learning
import fasttext

model =
fasttext.train_unsupervised(input="data/emotion/emotion_corpus_ukr.txt",
model="skipgram", epoch=100, lr=0.1, verbose=2, minCount=1, bucket=200000,
dim=100, minn=3, maxn=5)

```

```

with open('data/emotion/ukrtext.vec', 'w') as f:
    f.write(f"{len(model.get_words())} {model.get_dimension()}\n")
    for word in model.get_words():
        f.write(f"{word} {' '.join(map(str,
model.get_word_vector(word)))}\n")
#%%
import fasttext

# Train the model
model = fasttext.train_supervised(input="data/emotion/emotion_train_ukr.txt",
epoch=50, lr=0.1, verbose=2, minCount=1, bucket=200000, dim=100, minn=3,
maxn=5, pretrainedVectors="data/emotion/ukrtext.vec")

# sadness (0), joy (1), love (2), anger (3), fear (4), surprise (5).
id_to_label = {
    0: "sadness",
    1: "joy",
    2: "love",
    3: "anger",
    4: "fear",
    5: "surprise"
}

def print_results(model, input_path, k):
    num_records, precision_at_k, recall_at_k = model.test(input_path, k)
    f1_at_k = 2 * (precision_at_k * recall_at_k) / (precision_at_k +
recall_at_k)

    print("records\t{}".format(num_records))
    print("Precision@{}\t{:.3f}".format(k, precision_at_k))
    print("Recall@{}\t{:.3f}".format(k, recall_at_k))
    print("F1@{}\t{:.3f}".format(k, f1_at_k))
    print()

print_results(model, 'data/emotion/emotion_test_ukr.txt', k=1)

model.predict("Я дуже радий, що ти прийшов на мій день народження")

def calculate_accuracy(model, test_path, id_to_label):
    """
    Calculate accuracy for FastText model predictions.
    Returns overall accuracy and per-class accuracies.

```

```

"""
correct = 0
total = 0
class_correct = {i: 0 for i in id_to_label}
class_total = {i: 0 for i in id_to_label}

with open(test_path, 'r') as f:
    for line in f:
        # Split line into label and text
        parts = line.strip().split(' ', 1)
        if len(parts) != 2:
            continue

        true_label = int(parts[0].replace('__label__', ''))
        text = parts[1]

        # Get prediction
        pred = model.predict(text)[0][0]
        pred_label = int(pred.replace('__label__', ''))

        # Update counters
        total += 1
        class_total[true_label] += 1
        if pred_label == true_label:
            correct += 1
            class_correct[true_label] += 1

# Calculate accuracies
overall_accuracy = correct / total
class_accuracies = {id_to_label[i]: class_correct[i]/class_total[i]
                     for i in id_to_label if class_total[i] > 0}

# Print results
print(f"Overall Accuracy: {overall_accuracy:.3f}")
print("\nPer-class Accuracies:")
for emotion, acc in class_accuracies.items():
    print(f"{emotion}: {acc:.3f}")

return overall_accuracy, class_accuracies

# Usage with your existing model
calculate_accuracy(model, 'data/emotion/emotion_test_ukr.txt', id_to_label)

```

```

def count_fasttext_parameters(model):
    """
    Calculate the number of parameters in FastText model.
    """
    # Get model dimensions
    vocab_size = len(model.get_words())
    embedding_dim = model.get_dimension()
    output_dim = len(model.get_labels())

    # Calculate parameters
    embedding_params = vocab_size * embedding_dim
    output_layer_params = embedding_dim * output_dim
    total_params = embedding_params + output_layer_params

    # Print results
    print(f"Vocabulary size: {vocab_size:,}")
    print(f"Embedding dimension: {embedding_dim}")
    print(f"Number of classes: {output_dim}")
    print(f"\nParameters:")
    print(f"Embedding layer: {embedding_params:,}")
    print(f"Output layer: {output_layer_params:,}")
    print(f"Total parameters: {total_params:,}")

    return total_params

# Usage
total_params = count_fasttext_parameters(model)

model("hello my dear friend")

```

shap_analysis.py

```

from transformers import AutoTokenizer, AutoModelForSequenceClassification,
pipeline

# Load the tokenizer and model
tokenizer =
AutoTokenizer.from_pretrained("/Users/okorovii/Study/Emotion_classification_m
odel_e5_base")
model =
AutoModelForSequenceClassification.from_pretrained("/Users/okorovii/Study/Emo

```

```

tion_classification_model_e5_base")

detect_emotion = pipeline('text-classification', model=model,
tokenizer=tokenizer, top_k=None, device='cpu')
#%%
detect_emotion("Сьогодні знову не спав, чув вибухи десь поруч")
#%%
import shap

# explain the model on two sample inputs
explainer = shap.Explainer(detect_emotion)
shap_values = explainer(["Сьогодні знову не спав, чув вибухи десь поруч"])

# visualize the first prediction's explanation for the POSITIVE output class
shap.plots.text(shap_values[0, :, "FEAR"])

def pretty_print_shap(shap_value):
    """Pretty print SHAP values with tokens and their contributions."""
    print(f"Base value: {shap_value.base_values:.4f}\n")
    print(f"{'Token':<20} {'SHAP Value':<12} {'Impact':<10}")
    print("-" * 45)

    tokens = shap_value.data
    values = shap_value.values

    # Sort by absolute SHAP value (importance)
    sorted_indices = sorted(range(len(values)), key=lambda i: abs(values[i]),
reverse=True)

    for idx in sorted_indices:
        token = tokens[idx]
        value = values[idx]

        # Skip empty tokens
        if token.strip() == '':
            continue

        # Determine impact direction
        if value > 0:
            impact = "→ FEAR ✓"
        elif value < 0:
            impact = "← FEAR ✗"

```

```

        else:
            impact = "neutral"

    print(f"{token:<20} {value:>11.6f}  {impact}")

    final_score = shap_value.base_values + sum(values)
    print("-" * 45)
    print(f"Final prediction: {final_score:.4f}")

# Use the function
pretty_print_shap(shap_values[0, :, "FEAR"])

import transformers
import shap

# load a transformers pipeline model
model = transformers.pipeline('sentiment-analysis', return_all_scores=True)

# explain the model on two sample inputs
explainer = shap.Explainer(model)
shap_values = explainer(["What a great movie! ...if you have no taste."])

# visualize the first prediction's explanation for the POSITIVE output class
shap.plots.text(shap_values[0, :, "POSITIVE"])

```

train_transformer.py

```

import argparse
import json
import os

import evaluate
import torch
from datasets import load_dataset
from torch.optim import AdamW
from torch.utils.data import DataLoader
from tqdm.auto import tqdm
from transformers import (
    AutoModelForSequenceClassification,
    AutoTokenizer,
    DataCollatorWithPadding,
    get_scheduler,
)

```

```

def parse_args():
    parser = argparse.ArgumentParser(description="Simple text classification
training script")
    parser.add_argument("--model_name_or_path", type=str, required=True)
    parser.add_argument("--train_file", type=str, required=True)
    parser.add_argument("--validation_file", type=str, required=True)
    parser.add_argument("--output_dir", type=str, required=True)
    parser.add_argument("--max_length", type=int, default=128)
    parser.add_argument("--per_device_train_batch_size", type=int,
default=32)
    parser.add_argument("--per_device_eval_batch_size", type=int, default=32)
    parser.add_argument("--learning_rate", type=float, default=3e-4)
    parser.add_argument("--weight_decay", type=float, default=0.01)
    parser.add_argument("--num_train_epochs", type=int, default=10)
    parser.add_argument("--num_warmup_steps", type=int, default=0)
    parser.add_argument("--seed", type=int, default=42)
    return parser.parse_args()

def main():
    args = parse_args()
    torch.manual_seed(args.seed)

    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    os.makedirs(args.output_dir, exist_ok=True)

    extension = args.train_file.split(".")[-1]
    raw_datasets = load_dataset(
        extension,
        data_files={"train": args.train_file, "validation":
args.validation_file},
    )

    label_list = sorted(raw_datasets["train"].unique("label"))
    label_to_id = {label: i for i, label in enumerate(label_list)}
    num_labels = len(label_list)

    tokenizer = AutoTokenizer.from_pretrained(args.model_name_or_path)
    model = AutoModelForSequenceClassification.from_pretrained(
        args.model_name_or_path, num_labels=num_labels
    )

```

```

model.config.label2id = label_to_id
model.config.id2label = {i: label for label, i in label_to_id.items()}
model.to(device)

non_label_columns = [c for c in raw_datasets["train"].column_names if c
!= "label"]
text_column = non_label_columns[0]

def preprocess(examples):
    result = tokenizer(
        examples[text_column],
        padding=False,
        max_length=args.max_length,
        truncation=True,
    )
    result["labels"] = [label_to_id[l] for l in examples["label"]]
    return result

processed = raw_datasets.map(
    preprocess, batched=True,
remove_columns=raw_datasets["train"].column_names
)

collator = DataCollatorWithPadding(tokenizer)
train_loader = DataLoader(
    processed["train"],
    shuffle=True,
    batch_size=args.per_device_train_batch_size,
    collate_fn=collator,
)
eval_loader = DataLoader(
    processed["validation"],
    batch_size=args.per_device_eval_batch_size,
    collate_fn=collator,
)

no_decay = ["bias", "LayerNorm.weight"]
optimizer = AdamW(
    [
        {
            "params": [p for n, p in model.named_parameters() if not
any(nd in n for nd in no_decay)],
            "weight_decay": args.weight_decay,

```

```

        },
        {
            "params": [p for n, p in model.named_parameters() if any(nd
in n for nd in no_decay)],
            "weight_decay": 0.0,
        },
    ],
    lr=args.learning_rate,
)

max_train_steps = args.num_train_epochs * len(train_loader)
lr_scheduler = get_scheduler(
    name="linear",
    optimizer=optimizer,
    num_warmup_steps=args.num_warmup_steps,
    num_training_steps=max_train_steps,
)

metric = evaluate.combine(["precision", "recall", "f1"])
progress_bar = tqdm(range(max_train_steps))

for epoch in range(args.num_train_epochs):
    model.train()
    total_loss = 0.0
    for batch in train_loader:
        batch = {k: v.to(device) for k, v in batch.items()}
        outputs = model(**batch)
        loss = outputs.loss
        loss.backward()
        optimizer.step()
        lr_scheduler.step()
        optimizer.zero_grad()
        total_loss += loss.item()
        progress_bar.update(1)

    model.eval()
    for batch in eval_loader:
        batch = {k: v.to(device) for k, v in batch.items()}
        with torch.no_grad():
            outputs = model(**batch)
            preds = outputs.logits.argmax(dim=-1)
            metric.add_batch(predictions=preds, references=batch["labels"])
    eval_metric = metric.compute(average="micro")

```

```
        print(f"epoch {epoch}: loss={total_loss / len(train_loader):.4f}
{eval_metric}")

    model.save_pretrained(args.output_dir)
    tokenizer.save_pretrained(args.output_dir)
    with open(os.path.join(args.output_dir, "all_results.json"), "w") as f:
        json.dump({f"eval_{k}": v for k, v in eval_metric.items()}, f)

if __name__ == "__main__":
    main()
```

ДОДАТОК Б.

Акти впровадження результатів дисертаційного дослідження

ЗАТВЕРДЖУЮ

Проректор з навчальної роботи
Національного технічного університету України
«Київський політехнічний інститут
імені Ігоря Сікорського»
Тетяна ЖЕЛЯСКОВА
30 КВІ 2026 2026 р.

АКТ ВПРОВАДЖЕННЯ

результатів дисертаційного дослідження
асистента кафедри Системного програмування і спеціалізованих
комп'ютерних систем факультету програмних систем та прикладної математики
Коровія Олександра Сергійовича
за темою «Моделі та метод розпізнавання емоційної тональності фрагментів текстів в
універсальних комп'ютерних системах»
на здобуття освітньо-наукового ступеня доктора філософії
за спеціальністю 123 «Комп'ютерна інженерія»
освітньо-наукова програма Комп'ютерна інженерія
в освітній процес Національного технічного університету України
«Київський політехнічний інститут імені Ігоря Сікорського»

Комісія у складі:

голови комісії: Романкевича В. О. – завідувача кафедри Системного програмування і
спеціалізованих комп'ютерних систем, д.т.н., проф.;

членів комісії:

Терейковського І. А. – професора кафедри Системного програмування і
спеціалізованих комп'ютерних систем, д.т.н., проф.;

Клятченка Я. М. – доцента кафедри Системного програмування і спеціалізованих
комп'ютерних систем, к.т.н., доц.

засвідчує, що результати дисертаційного дослідження Коровія Олександра Сергійовича на
тему «Моделі та метод розпізнавання емоційної тональності фрагментів текстів в
універсальних комп'ютерних системах», а саме: підхід до оцінювання ефективності
нейромережових засобів розпізнавання емоційного забарвлення тексту, метод розпізнавання
емоційного забарвлення фрагментів тексту на основі моделі побудови нейромережових
засобів, адаптованих до лінгвістичної специфіки української мови та обмежених
обчислювальних ресурсів, впроваджено у лекційні та практичні заняття з навчальної
дисципліни «Комп'ютерні системи штучного інтелекту» за спеціальністю 123 «Комп'ютерна
інженерія» освітньо-наукової програми «Комп'ютерна інженерія»:

Лекція 6. Глибинні рекурентні нейронні мережі.

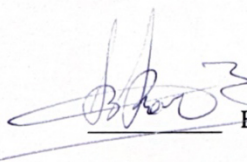
Лекція 9. Метод нейромережевого аналізу текстових даних.

Лабораторне заняття 3. Розробка та дослідження програмного модулю визначення емоційного забарвлення тексту

Посилання на силабус: «https://scs.kpi.ua/wp-content/uploads/silabys/2025-2026/Б/4/nSilabus_CSAI_2025_26.docx.pdf».

Зазначене актуалізує зміст навчальної дисципліни до сучасних вимог галузі, розкриває методи побудови засобів розпізнавання ключових слів, адаптованих до роботи в умовах обмежених обчислювальних ресурсів, та демонструє практичне застосування критеріїв оцінювання їх ефективності.

Завідувач кафедри
Системного програмування і
спеціалізованих комп'ютерних систем,
доктор технічних наук, професор



Віталій РОМАНКЕВИЧ

Професор кафедри
Системного програмування і
спеціалізованих комп'ютерних систем,
доктор технічних наук, професор



Ігор ТЕРЕЙКОВСЬКИЙ

Доцент кафедри
Системного програмування і
спеціалізованих комп'ютерних систем,
кандидат технічних наук, доцент



Ярослав КЛЯТЧЕНКО

ЗАТВЕРДЖУЮ

Проректор з наукової роботи
Національного технічного університету
України

«Київський політехнічний інститут
імені Ігоря Сікорського»

Сергій СТРЕНКО

« 30 »

2026 р.

А К Т

про використання результатів дисертаційного дослідження старшого викладача кафедри Системного програмування і спеціалізованих комп'ютерних систем факультету прикладної математики Національного технічного університету України «КПІ імені Ігоря Сікорського» Коровія Олександра Сергійовича на тему «Моделі та метод розпізнавання емоційної тональності фрагментів текстів в універсальних комп'ютерних системах» на здобуття освітньо-наукового ступеня доктора філософії.

Комісія у складі: голова – декан факультету прикладної математики «КПІ ім. Ігоря Сікорського», д.т.н., проф. Дичка І.А.; члени комісії – професор кафедри СПіСКС КПІ ім. Ігоря Сікорського, д.т.н., проф. Романкевич О.М.; професор кафедри СПіСКС КПІ ім. Ігоря Сікорського, д.т.н., проф. Терейковський І.А. цим Актом засвідчує, що результати дисертаційної роботи Коровія Олександра Сергійовича отримані ним особисто та використані у:

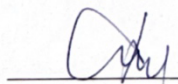
- НДР «Методи, моделі та засоби моніторингу закликів до тероризму у онлайн соціальних мережах», що виконується за ініціативою авторів, № держреєстрації 0124U003866, закінчення у 2026 р.

В НДР використані наступні результати:

- Метод розпізнавання емоційної тональності фрагментів тексту на основі моделі побудови нейромережових засобів, адаптованих до лінгвістичної специфіки української мови та обмежених обчислювальних ресурсів.

- Архітектуру програмної системи розпізнавання емоційної тональності, відповідні програмні застосунки та спеціалізований анований корпус україномовних текстів.

Голова комісії
д.т.н., проф.



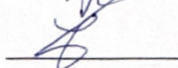
Іван ДИЧКА

Члени комісії
д.т.н., проф.



Олексій РОМАНКЕВИЧ

д.т.н., проф.



Ігор ТЕРЕЙКОВСЬКИЙ