

ВЕБ-ЗАСТОСУНОК ДЛЯ АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТУ

WEB APPLICATION FOR SENTIMENT ANALYSIS

Яскулко Д.Д.,

*orcid.org/0009-0000-7875-5858**студентка I курсу магістратури філологічного факультету
Харківського національного університету імені В.Н. Каразіна*

Дубовик Н.М.,

*orcid.org/0000-0001-5748-1296**кандидат філологічних наук, доцент,
завідувач кафедри загального та прикладного мовознавства
Харківського національного університету імені В.Н. Каразіна*

Стаття присвячена актуальній для сучасного мовознавства темі – обробці текстової інформації. У наш час, коли інформаційні технології розвиваються з неймовірною швидкістю, а міждисциплінарність стає основою наукових досягнень, обробка великих обсягів текстових даних набуває особливої значущості. Лінгвістичні наукові розвідки, в яких залучається статистична обробка, потребують застосування новітніх методів та інструментів. Інтеграція сучасних комп'ютерних технологій у лінгвістику не лише покращує якість наукових досліджень, але й робить їх результативнішими та адаптованими до потреб сучасного суспільства. Завдяки цьому розвивається прикладне мовознавство, яке активно впливає на розвиток технологій, освіти та інших сфер.

Наше дослідження спрямоване на розробку веб-застосунку для автоматичного аналізу тональності тексту, який дозволяє визначати емоційне забарвлення тексту (позитивне, негативне або нейтральне). Реалізований веб-застосунок базується на клієнт-серверній архітектурі, яка забезпечує ефективну взаємодію між фронтендом і бекендом. На клієнтському рівні створено простий та зручний інтерфейс для введення тексту та відображення результатів аналізу користувачеві. Серверний рівень відповідає за попередню обробку тексту, визначення тональності та повернення результатів через API.

Для аналізу тональності тексту застосовувався підхід на основі правил із використанням інструмента VADER, який визначає тональність тексту. Для попередньої обробки текстової інформації використовувалася бібліотека NLTK, яка виконує необхідні операції для нормалізації тексту, які є важливими кроками для забезпечення високої точності подальшого аналізу. У статті детально розглянуто всі етапи розробки веб-застосунку: від визначення функціонального призначення до тестування. Результати дослідження показують, що розроблене програмне забезпечення для автоматичного аналізу тональності тексту має значний потенціал для застосування в різних сферах, від лінгвістичних досліджень до аналізу медіа-контенту.

Ключові слова: обробка природної мови, аналіз тональності, прикладна лінгвістика, веб-застосунок, Flask, VADER.

The paper is devoted to the relevant topic to modern linguistics – text information processing. Nowadays, when information technologies are developing at an incredible speed and interdisciplinarity becomes the foundation of scientific achievements, processing large volumes of textual data has gained special significance. Linguistic research, which involves statistical processing, requires the application of the latest methods and tools. The integration of modern computer technologies into linguistics not only improves the quality of scientific research but also makes them more effective and adaptable to the needs of modern society. Thanks to this, applied linguistics is developing, which has a significant impact on the development of technology, education and other fields.

Our study is aimed at developing the web application for automatic sentiment analysis, which allows determining the emotional tone of the text (positive, negative or neutral). The implemented web application is based on a client-server architecture, which ensures efficient interaction between the frontend and the backend. On the client side, a simple and convenient interface has been created for text input and displaying the analysis results to the user. The server side is responsible for preprocessing the text, determining the sentiment and returning the results via an API.

A rule-based approach was used for sentiment analysis, utilizing the VADER tool, which determines the tonality of the text. For preprocessing the textual data, the NLTK library was employed, which performs the necessary operations for text normalization, which are important steps for ensuring high accuracy of subsequent analysis. The article provides a detailed overview of all stages of the web application development, from defining the functional purpose to testing. The research results show that the developed software for automatic sentiment analysis has significant potential for application in various fields, from linguistic studies to media content analysis.

Key words: natural language processing, sentiment analysis, applied linguistics, web application, Flask, VADER.

Постановка проблеми. Мовознавчі розвідки завжди передбачали роботу із текстами, у сучасних умовах розвитку цифрових технологій цей процес набагато полегшується для дослідника.

Завдання сучасного мовознавця полягає не лише у тому, що зібрати необхідний масив даних, але й якомога швидше і якісніше проаналізувати отриману інформацію, доки не втрачена акту-

альність досліджуваного матеріалу. Обробка текстової інформації передбачає часто значні обсягом масиви, що вимагає оптимізації процесу, яка стає можливою завдяки цифровим технологіям. Відповідно, машинний аналіз тексту набуває все більшого значення, оскільки він є важливим інструментом для обробки великих обсягів даних та автоматизації різноманітних завдань.

Sentiment analysis, або **аналіз тональності тексту** – один із основних, на нашу думку, напрямів комп'ютерної лінгвістики та обробки природної мови. Його метою є визначати полярність тексту, тобто яким є текст – *позитивним, негативним або нейтральним* [12]. Він широко використовується для аналізу відгуків клієнтів, соціальних медіа-дописів, новин, коментарів тощо. Такий аналіз має очевидну перспективу для обробки будь-якої текстової інформації у лінгвістичних наукових дослідженнях.

Аналіз останніх досліджень і публікацій. Останні дослідження у галузі обробки природної мови показують, що **аналіз тональності тексту** є одним із ключових завдань, яке спрямоване на класифікацію емоційного забарвлення тексту, і стає актуальним у контексті зростаючих обсягів текстових даних у цифровому середовищі. Зокрема, відзначаємо роботи, присвячені аналізу тональності тексту, які зосереджуються на методах аналізу тональності в системах аналізу опіній користувачів соціальних мереж [1], а також на аналізі тональності тексту українською мовою [2]. Ці праці підкреслюють важливість та актуальність використання такого аналізу в мовознавчих та комп'ютерних дослідженнях.

Завдяки розвитку *алгоритмів машинного навчання, обчислювального інтелекту та інструментів аналізу тексту*, можливості автоматизованого аналізу емоційної складової текстів значно розширилися. Новітні технології дозволяють не лише класифікувати текст на позитивний, нейтральний чи негативний, а й виявляти складніші аспекти, такі як сарказм, іронія або багатозначність емоційного змісту. У сучасних дослідженнях виокремлюють три основні підходи до визначення тональності тексту:

- методи на основі правил (Rule-based methods);
- методи машинного навчання (Machine learning methods);
- гібридні методи (Hybrid methods) [11].

Методи на основі правил. Підхід на основі правил полягає у використанні заздалегідь визначених правил та словників, які містять слова з відповідними емоційними оцінками. Одним із популярних інструментів цього підходу є **VADER**,

який використовує лексикон слів і оцінки їх настрою, щоб визначити загальну тональність тексту [9].

Методи машинного навчання. Підхід машинного навчання застосовує алгоритми, які навчаються на маркованих наборах даних для автоматичної класифікації емоційного забарвлення тексту. Серед поширених алгоритмів є найпростіший класифікатор Байєса, логістична регресія, глибоке навчання (**DL**) та опорні вектори (**SVM**) [13].

Гібридні методи. Гібридний підхід поєднує переваги методів на основі правил і машинного навчання. На початковому етапі обробка тексту виконується за допомогою правил та лексиконів, які мають відповідні емоційні оцінки. Після цього застосовуються алгоритми машинного навчання для більш глибокого аналізу, які враховують контекстуальні залежності між словами, багатозначність і складні мовні конструкції. Такий підхід забезпечує високу точність і універсальність, що робить ефективним для аналізу різноманітних текстових даних [8].

Постановка завдання. Метою статті є опис розробки веб-застосунку, який дозволяє визначати тональність тексту, введеного користувачем та відображати результати обробки.

Виклад основного матеріалу. Розробка веб-застосунку – це процес створення програмного забезпечення, яке працює на веб-сервері та доступне користувачам через веб-браузер. Веб-застосунки базуються на клієнт-серверній архітектурі, де користувачі взаємодіють із інтерфейсом у браузері, а серверна частина відповідає за обробку запитів і виконання основних функцій застосунку. Реалізація веб-застосунку для аналізу тональності тексту складалася з 6 етапів:

- визначення функціонального призначення веб-застосунку;
- створення дизайну;
- вибір технологій;
- розробка інтерфейсу користувача;
- розробка серверної частини;
- тестування.

Кожен етап має своє важливе значення і потребує коректної реалізації. Звернемося до опису зазначеного процесу.

Визначення функціонального призначення веб-застосунку. Перший етап розробки веб-застосунку полягає у визначенні його основної мети та функцій, які він має виконувати. На цьому етапі важливо зрозуміти, для кого розробляється веб-застосунок, які задачі він має вирішувати, і як вони будуть реалізовані. Це дозволяє створити

чітке технічне завдання, яке стане основою для подальшої розробки. Реалізований веб-застосунок призначений для автоматичного визначення емоційної тональності тексту. За допомогою нього користувачі можуть визначити, яким є текст – *позитивним, негативним* або *нейтральним*.

Створення дизайну. Після визначення функціонального призначення йде етап створення дизайну. **Дизайн** є важливим етапом, оскільки він визначає не лише естетичний вигляд веб-застосунку, а й зручність взаємодії користувачів із програмним забезпеченням. Під час розробки дизайну створюється детальний макет інтерфейсу, який відображає основні елементи – кнопки, поля для введення, меню та інші елементи. Для проектування дизайну використовувався редактор **Figma** – інструмент для створення макетів і прототипів інтерфейсів [5].

Вибір технологій. На етапі вибору технологій визначаються мови програмування, фреймворки, бібліотеки та інші інструменти, які будуть використані для реалізації веб-застосунку. Для розробки фронтенду (інтерфейсу користувача) були обрані такі технології, як **HTML** та **CSS**, для бекенду (серверної частини) – фреймворк **Flask**, а для аналізу тональності тексту – бібліотека **NLTK** та інструмент **VADER**.

Розробка інтерфейсу користувача. Після обрання необхідних технологій розпочинається розробка фронтенду, що включає в себе створення веб-сторінок, компонентів інтерфейсу, а також інтерактивних елементів. Для реалізації інтерфейсу користувача було обране середовище **Visual Studio Code**, яке використовується для розробки сайтів і застосунків [10]. У ньому був створений файл під назвою **page.html**, який містить у собі мову розмітки **HTML** – для створення сторінки та мову стилів **CSS** – для її зовнішнього вигляду. За допомогою цих технологій ми створили зручний у використанні та привабливий інтерфейс веб-застосунку. Нижче подано рисунок головної сторінки веб-застосунку, на якому зображений заголовок *Sentiment analysis*, текстове поле для введення тексту або речення лише англійською мовою, кнопка **Processing** для визначення тональності тексту, ім'я та прізвище розробника та рік створення веб-застосунку (рис. 1).

Розробка серверної частини. Після розробки інтерфейсу користувача наступним етапом є реалізація серверної частини (бекенду), яка відповідає за виконання основних функцій веб-застосунку та обробку користувацьких запитів. Вона отримує запити користувача, здійснює необхідну обробку даних та передає результат на фронтенд

для відображення. Серверна частина побудована з використанням фреймворка **Flask** в інтегрованому середовищі розробки **PyCharm**, яке надає розробникам різноманітні інструменти для програмування на **Python** [3]. У ньому був створений файл під назвою **web-app.py**, який містить у собі:

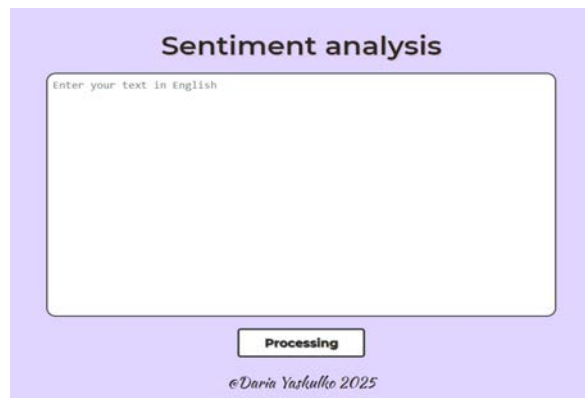


Рис. 1. Головна сторінка веб-застосунку

– **Flask** – фреймворк для створення веб-застосунків на мові програмування Python, що надає можливість створювати маршрути, обробляти запити від користувачів і інтегрувати їх з **HTML**-шаблонами [4].

– **NLTK** – бібліотека для обробки природної мови на Python. Завдяки своїй багатofункціональності, вона активно використовується для аналізу тональності тексту, класифікації, автоматичного перекладу та інших завдань у сфері NLP [6].

– **VADER** – інструмент для автоматичного аналізу настроїв тексту, який використовує лексикон настроїв для визначення тональності тексту. Під час аналізу тексту **VADER** розбиває його на окремі слова та перевіряє кожне слово за лексиконем настроїв [7].

Функціонал веб-застосунку полягає в тому, що користувач вводить англійський текст або речення у текстове поле, після чого натискає кнопку **Processing**. Сервер здійснює обробку тексту, використовуючи бібліотеку **NLTK** та інструмент **VADER**. На основі результатів аналізу, система повертає користувачеві відповідь про тональність тексту, який може бути *позитивним, негативним* або *нейтральним*.

Тестування. Останнім етапом після розробки фронтенду та бекенду є тестування веб-застосунку, яке є необхідною частиною процесу забезпечення якості. Завдяки тестуванню можна виявити помилки та недоліки у програмному забезпеченні до моменту його впровадження. Тестування дозволяє переконатися в тому, що система функціо-

нує належним чином, не містить помилок і повністю відповідає потребам користувачів.

Для користування веб-застосунком потрібно запуснути файл **web-app.py** та перейти за відповідним портом (рис. 2).

* Running on <http://127.0.0.1:5000>

Рис. 2. Порт для користування веб-застосунком

Після переходу на головну сторінку (рис. 1) користувач вводить у текстове поле текст або речення англійською мовою, наприклад, *A Front-End Developer is someone who creates websites and web applications*, потім натискає на кнопку **Processing** і отримує відповідний результат обробки (рис. 3).

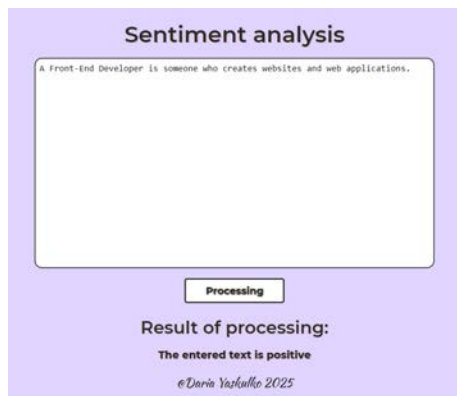


Рис. 3. Результат обробки

Після обробки введеного тексту або речення, на сторінці буде відображатися заголовок **Sentiment analysis**, текстове поле, кнопка **Processing**, заго-

ловок **Result of processing**, а також рядок із результатом аналізу **The entered text is positive/negative/neutral**. Додатково на сторінці буде вказано ім'я та прізвище розробника та рік створення веб-застосунку.

Висновки. Підводячи підсумки нашої наукової розвідки, маємо зазначити, що дослідження має теоретичну та практичну цінність. Адже описує як процес і етапи створення веб-застосунку, так і його практичну реалізацію. Розробка веб-застосунку для аналізу тональності тексту є корисним програмним забезпеченням для автоматичного визначення емоційного забарвлення тексту. Завдяки використанню сучасних технологій та інструментів, нам вдалося реалізувати функціональний та зручний веб-застосунок. Цей веб-застосунок є прикладом успішного поєднання сучасних технологій, інженерних рішень та зручності користувацького досвіду, що забезпечує високу якість автоматичного аналізу тональності тексту.

Перспектива подальшого дослідження полягає у вдосконаленні веб-застосунку для аналізу тональності тексту завдяки інтеграції новітніх технологій штучного інтелекту. Використання трансформерів та нейронних мереж дозволить значно покращити точність визначення тональності текстів. Одним із важливих напрямів розвитку є розширення мовної підтримки, яка надасть можливість здійснювати багатомовний аналіз текстів. З розвитком технологій глибокого навчання можна також очікувати підвищення швидкості аналізу та масштабованості системи, що забезпечить її здатність обробляти великі обсяги текстових даних у реальному часі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

1. Кульчицький А.Б., Турко Ю.В. Методи аналізу тональності тексту в системах аналізу опіній користувачів соціальних мереж. СІТ'2020, Тернопіль, 30 листопада 2020. С. 53–54. URL: <http://zfs-journal.uzhnu.uz.ua/8302-2015.pdf> (дата звернення: 28.01.2025).
2. Рябишев О.В., Єрохін А.Л., Бахмет А.Г. Аналіз тональності тексту українською мовою. *Науково-технічний журнал «Біоніка інтелекту»*. 2021 № 1 (96). С. 15–21. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/f59561cb-66a5-4f99-b6ff-cf30a5162f91/content> (дата звернення: 19.01.2025).
3. Використання середовища розробки Pycharm при вивченні програмування: веб-сайт. URL: https://informatika.udpu.edu.ua/?page_id=7591 (дата звернення: 22.01.2025).
4. Вступ до Python фреймворків: які вони бувають і для чого використовуються. *FoxmindEd.*: веб-сайт. URL: <https://foxminded.ua/freimvorky-python/> (дата звернення: 20.01.2025).
5. Figma: що це таке, для чого і кому потрібна. *GoIT.*: веб-сайт. URL: <https://goit.global.ua/articles/figma-shcho-tse-take-dlia-choho-i-komu-potribna/> (дата звернення: 18.01.2025).
6. NLTK Sentiment Analysis Tutorial for Beginners. *Moez Ali.*: веб-сайт. URL: <https://www.datacamp.com/tutorial/text-analytics-beginners-nltk> (дата звернення: 26.01.2025).
7. Sentiment Analysis in Python Using VADER. *Editorial Team.*: веб-сайт. URL: <https://towardsai.net/p//sentiment-analysis-in-python-using-vader> (дата звернення: 22.01.2025).
8. Top 5 Techniques for Sentiment Analysis in Natural Language Processing. *Syed Huma Shah.*: веб-сайт. URL: <https://medium.com/illumination/top-5-techniques-for-sentiment-analysis-in-natural-language-processing-c07ba5b83f64> (дата звернення: 24.01.2025).

9. Vader: A Comprehensive Guide to Sentiment Analysis in Python. *Lavanya Geetha.*: веб-сайт. URL: <https://medium.com/@rslavanyageetha/vader-a-comprehensive-guide-to-sentiment-analysis-in-python-c4f1868b0d2e> (дата звернення: 04.01.2025).

10. Visual Studio Code. *VOLL.*: веб-сайт. URL: <https://voll.com.ua/uk/glossary/visual-studio-code> (дата звернення: 20.01.2025).

11. What Is Sentiment Analysis? *Eric Kleppen.*: веб-сайт. URL: <https://builtin.com/machine-learning/sentiment-analysis> (дата звернення: 07.01.2025).

12. What is Sentiment Analysis? *Geeksforgeeks.*: веб-сайт. URL: <https://www.geeksforgeeks.org/what-is-sentiment-analysis/> (дата звернення: 09.01.2025).

13. What is sentiment analysis? *IBM.*: веб-сайт. URL: <https://www.ibm.com/think/topics/sentiment-analysis#:~:text=Sentiment%20analysis%20uses%20natural%20language,the%20two%20known%20as%20hybrid.> (дата звернення: 15.01.2025).