

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет

Центр заочної, дистанційної та вечірньої форм навчання

Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

Ігор

ШЕЛЕХОВ

_____ (підпис)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня магістр

зі спеціальності 122 – Комп'ютерних наук,

освітньо-професійної програми «Інформатика»

на тему: «Інформаційна технологія інтелектуального аналізу тональності тексту»

здобувачки групи ІН.мз-21с Панченко Аліни Анатоліївни

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

Аліна ПАНЧЕНКО

_____ (підпис)

Керівник

В.о. завідувача кафедри

кандидат технічних наук/кандидат

Ігор ШЕЛЕХОВ

_____ (підпис)

Суми – 2023

Сумський державний університет
Центр заочної, дистанційної та вечірньої форм навчання
Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

(підпис)

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»

здобувачки групи ІН.мз-21с Панченко Аліни Віталіївни

1. Тема роботи: «Інформаційна технологія інтелектуального аналізу тональності тексту» затверджую наказом по СумДУ від «20» листопада 2023 р. № 1308-VI
2. Термін здачі здобувачем кваліфікаційної роботи до 14 грудня 2023 року
3. Вхідні дані до кваліфікаційної роботи _____
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)
1) Аналіз проблеми предметної області, постановка й формування завдань дослідження. 2) Огляд використаних технологій. 3) Реалізація чат боту, моделювання програмного забезпечення. 4) Аналіз отриманих результатів
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
6. Консультанти до проєкту (роботи), із значенням розділів проєкту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання « 25 » вересня 2023 р.

Завдання прийняв до виконання _____

(підпис)

Керівник _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	<i>Аналіз проблеми предметної області, постановка й формування завдань дослідження</i>	15.10-25.10	
2	<i>Огляд використаних технологій</i>	25.10-30.10	
3	<i>Реалізація чат боту, моделювання програмного забезпечення</i>	30.10-15.11	
4	<i>Аналіз отриманих результатів</i>	15.11-20.11	
5	<i>Оформлення пояснювальної записки до кваліфікаційної роботи</i>	20.11-01.12	

Здобувач вищої освіти _____

(підпис)

Керівник _____

(підпис)

АНОТАЦІЯ

Записка: 54 сторінок, 29 рисунків, 20 таблиць, 1 додаток, 14 джерел.

Обґрунтування актуальності теми роботи — Аналіз тональності тексту є актуальною проблемою контент-аналізу інформаційних джерел в комп'ютерній лінгвістиці, що дозволяє визначити характер їх емоційної забарвленості та ставлення авторів.

Об'єктом дослідження є процес аналізу емоційної забарвленості англomовних текстів.

Предметом дослідження є порівняння методів аналізу настроїв у текстах, включаючи згорткові нейронні мережі, рекурентні нейронні мережі та двонаправлені трансформери Берта.

Мета роботи: розробка телеграм бота з використанням попередньо навченої моделі для класифікації тексту.

Результати дослідження: здійснено огляд методів класифікації тексту та обрано попередньо навчену модель для обробки тональності тексту. Також її було імплементовано в телеграм бот.

ШТУЧНИЙ ІНТЕЛЕКТ, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, РЕКУРЕНТНІ
НЕЙРОННІ МЕРЕЖІ, ДВОНАПРАВЛЕНИЙ ТРАНСФОРМЕР BERT
КЛАСИФІКАЦІЯ ТЕКСТУ.

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1. АКТУАЛЬНІСТЬ ТЕМИ І МЕТА ДОСЛІДЖЕННЯ	5
1.1 Актуальність теми та мета дослідження	5
1.2 Аналіз предметної області	6
1.3 Огляд готових рішень	7
1.4 Методи задач обробки природної мови	9
1.4.1 Методи класифікації тональності та класифікація текстів	11
1.5 Порівняльна характеристика та формули розрахунків точності	15
1.6 Порівняльний аналіз алгоритмів CNN,BERT,RNN	16
1.6.1 Двонаправлена нейронна мережа кодувальник - BERT	16
1.6.2 Згорткова нейронна мережа	19
1.6.3 Рекурентна нейронна мережа	20
1.6.4 Порівняння моделі RoBERTa з іншими модифікаціями методу BERT	27
Висновки до розділу 1	32
РОЗДІЛ 2 РЕАЛІЗАЦІЯ ЧАТ-БОТА	33
2.1 Використані технології	33
2.2 Опис методів	33
2.3 Підключення моделі	34
2.4 Опис програмного продукту	35
2.5 Моделювання програмного забезпечення	35
2.5.1 Діаграма використання	35
2.5.2 BPMN	36
2.5.3 Діаграма активності	37
Висновки до розділу 2	39
РОЗДІЛ 3. ТЕСТУВАННЯ ТЕЛЕГРАМ БОТА	40

3.1	Опис методу тесуванн	40
3.2	Приклад використання	45
	Висновки до розділу 3	53
	ВИСНОВКИ	54
	ВИКОРИСТАНІ ДЖЕРЕЛА	56
	ДОДАТОК А	58

ВСТУП

Обґрунтування вибору теми роботи.

Програмісти, студенти та звичайні люди, які цікавляться ІТ-сектором, часто обговорюють обробку природної мови, як і наукове співтовариство, де ці методи є особливо важливими, оскільки вони забезпечують основу для створення систем штучного інтелекту.

Актуальність.

Підхід, який прагне виявити тонкі емоційні відтінки в усіх видах літературних творів, — це аналіз настрою, який є найзахоплюючішим прийомом. Через експоненціальне зростання кількості користувачів Інтернету та кількості часу, який вони проводять у групах новин, соціальних мережах тощо, аналіз настроїв в Інтернеті стає все більш важливим.

Оскільки автоматизований аналіз даних про певні продукти чи послуги став простішим і ефективнішим, дуже важливо оцінити тон тексту.

Крім того, що аналіз тону тексту має вирішальне значення для маркетингу брендів, дослідження ринку, галузевих тенденцій і вивчення робочої сили — усе це має вирішальне значення в сучасному світі — аналіз тону тексту може бути корисним для інвестування в акції, моніторингу кредитного ринку та інших важливих завдань. З огляду на те, що існує кілька видатних і корисних рішень цієї проблеми, має сенс оцінити складні алгоритми на швидкість, якість і точність.

Крім того, майте на увазі, що наразі в Telegram немає інструментів, які дозволяють перевіряти повідомлення на основі їх тональності. Тому має сенс базувати бота на алгоритмі, який має найкращу попередньо навчену модель.

Предметом дослідження є порівняння методів аналізу настроїв у текстах, включаючи згорткові нейронні мережі, рекурентні нейронні мережі та двонаправлені трансформери Берта.

Об'єктом дослідження є процес аналізу емоційної забарвленості англomовних текстів..

Гіпотеза. Аналіз іноземних текстів щодо їх емоційного забарвлення дозволяє надати додаткову інформацію для їх розуміння читачами, які не є носіями відповідної мови.

Новизна. В роботі виконано порівняльний аналіз методів і моделей класифікації тональності текстів.

Структура. Дане робота складається зі вступу, аналітичного огляду, постановки задачі, вибір методу розв'язання поставленої задачі, опису програмного забезпечення інформаційної системи, висновків, списку використаних джерел та додатків.

1 АКТУАЛЬНІСТЬ ТЕМИ І МЕТА ДОСЛІДЖЕННЯ

1.1 Аналіз предметної області

Щоб визначити загальні настрої (позитивні, негативні чи нейтральні) щодо певних предметів, товарів чи послуг, попередньо підготовлені моделі можуть вивчати дописи та коментарі користувачів у соціальних мережах. Сьогодні люди будь-якого віку досить часто користуються соціальними мережами, тому що вони значно полегшують спілкування, ділитися оцінками та висловлювати думки в Інтернеті. Сьогодні більшість людей вважають за краще робити покупки онлайн, і, як наслідок, вони також пишуть відгуки онлайн, що допомагає підприємствам збирати дані та оцінювати задоволеність і настрої клієнтів. Розглянемо випадок, коли споживач робить онлайн-покупку, а потім отримує пошкоджений продукт. Ігнорування проблеми може призвести до великого поширення інформації в соціальних мережах і завдати шкоди репутації компанії чи бренду.

У результаті таких обставин може виникнути важливий аргумент, але ви можете розсіяти його або, загалом, вжити заходів, щоб уникнути таких обставин, оцінивши ставлення покупця.

Аналіз настроїв виявляє як позитивні, так і негативні відгуки, даючи компаніям зрозуміти, як громадськість ставиться до їх бренду. Дослідження в цій галузі можуть дати інформацію, яку можна використати для оцінки кризових сценаріїв, порівняння компанії з конкурентами та запобігання можливим конфліктам до їх виникнення.

Знайти товари чи послуги, які найбільше подобаються чи користуються попитом, може допомогти аналіз настроїв. Алгоритми аналізу тональності повністю завершують роботу. Маркетологи можуть прогнозувати майбутні тенденції та допомагати керівництву приймати обґрунтовані рішення шляхом постійного моніторингу та оцінки моделей поведінки клієнтів.[4]

Перевага цих методів полягає в тому, що вони прості у застосуванні, оскільки штучний інтелект збирає та аналізує важливі дані, не вимагаючи великої кількості джерел. Враховуючи все, використання попередньо підготовлених моделей для

аналізу настроїв тексту може дати глибоку інформацію про поведінку та вподобання споживачів, дозволяючи підприємствам краще реагувати на зміни ринку та змінювати свої маркетингові плани.

1.2 Огляд готових рішень

Згадки про бренди відстежуються, а тон тексту оцінюється Mediatoolkit, онлайн-інструментом моніторингу соціальних мереж.

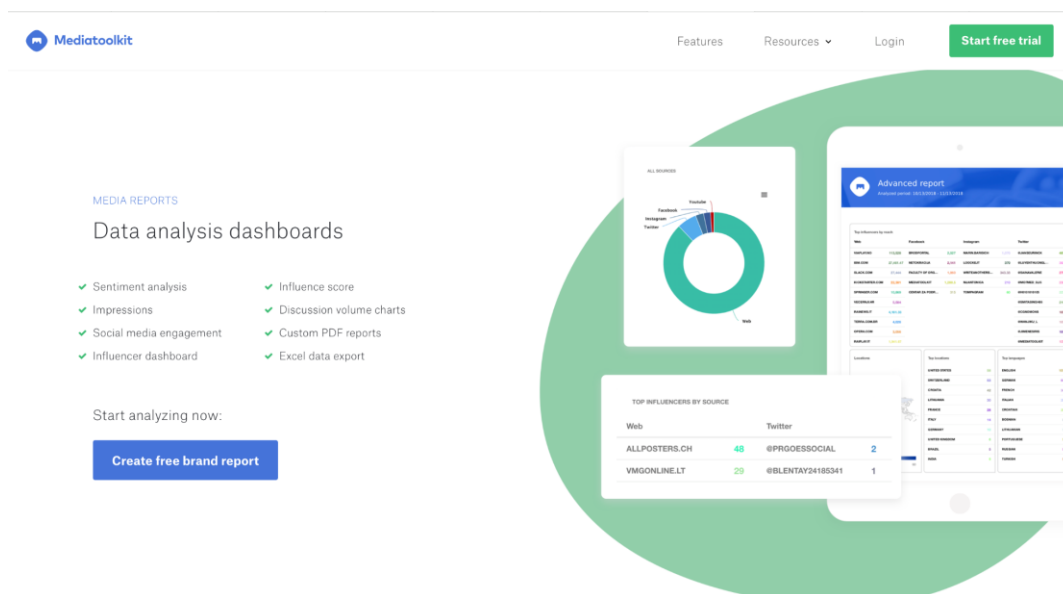


Рисунок 1.1 - Mediatoolkit

У 2014 році була заснована лондонська компанія TEXT2DATA, що розробляє програмне забезпечення для текстової аналітики. Компанії надають аналіз настроїв для широкого діапазону текстових форматів. Навіть онлайн ви можете навчити свою власну модель.

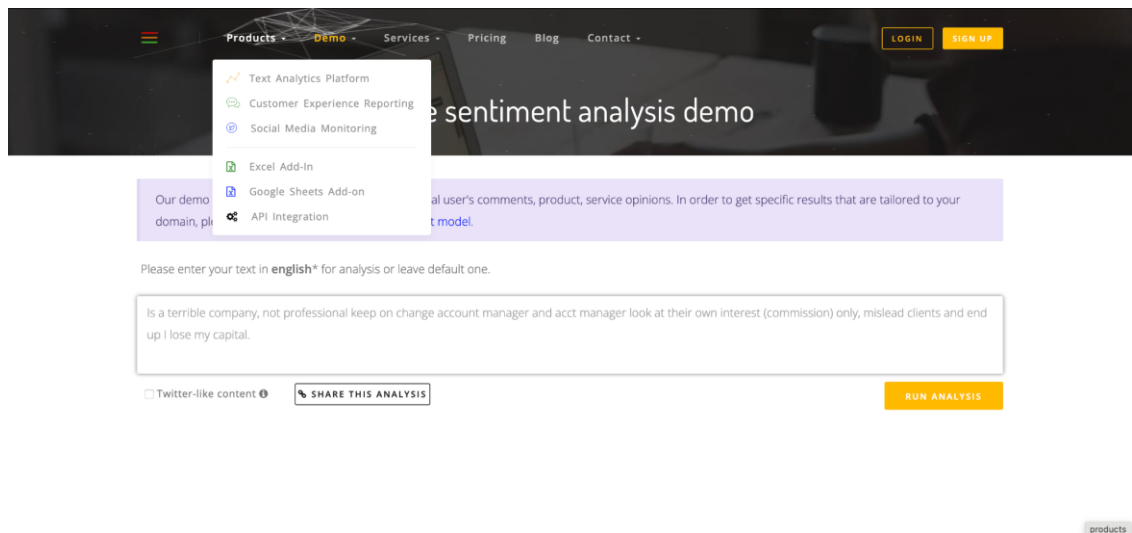


Рисунок 1.2 - Text2Data

Talkwalker — це інструмент аналізу настроїв, який вивчає думки про ваш бізнес на платформах соціальних мереж.

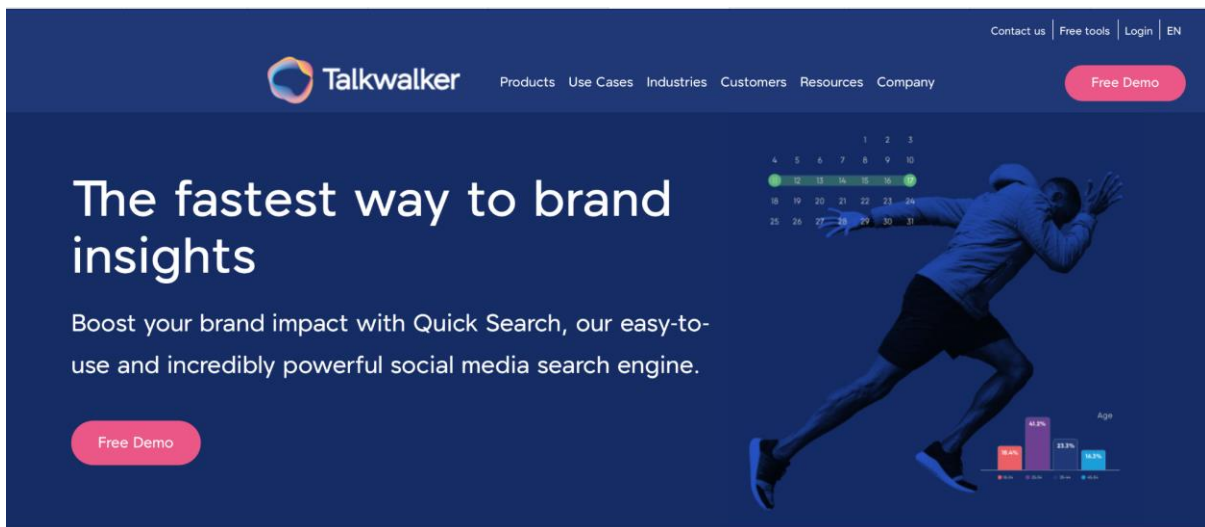


Рисунок 1.3 - Talkwalker

Critical Mention відрізняється від інших представлених програм тим, що переглядає новини та інший вміст, пов'язаний з вашим бізнесом. Ви можете стежити за всіма розмовами, що стосуються вашого бренду, і оперативно вирішувати будь-які суперечки, які можуть виникнути через цілодобовий аналіз.

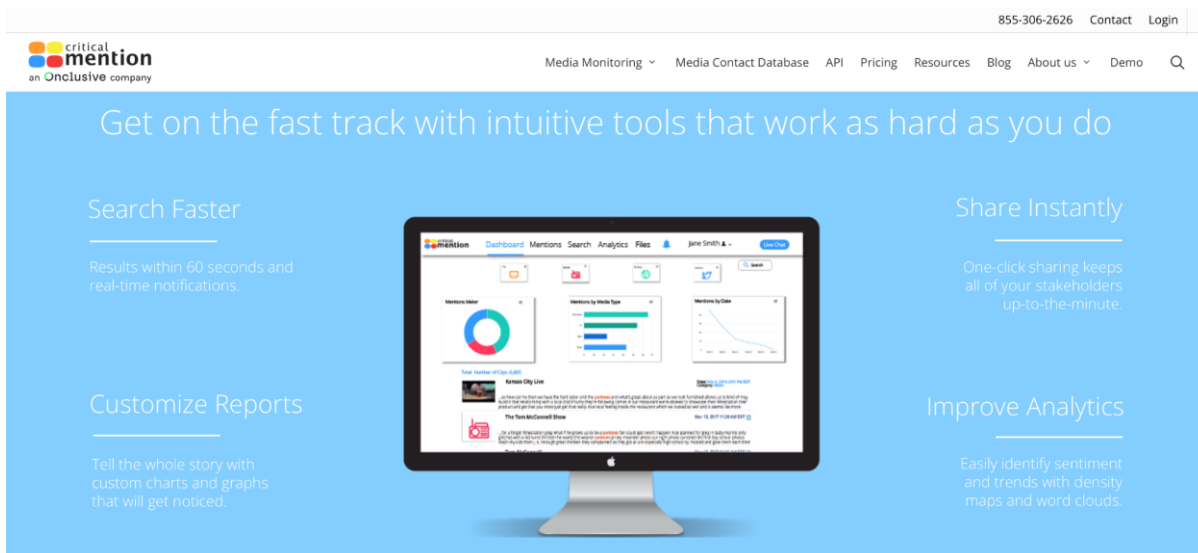


Рисунок 1.4 - Critical Mention

Завдання аналізу настроїв вирішується програмним забезпеченням, яке представлено; однак це платна послуга, яка зазвичай коштує 300 доларів на місяць. З цієї причини я вважаю доречним розробити безкоштовного бота Telegram.

1.3 Методи задач обробки природної мови

Задачі обробки природної мови мають в більшості випадків одновимірний простір, а саме розділення на два випадки:

1. позитив
2. негатив.

Однак це не означає, що шкала оцінювання не може бути змінена; фактично, цілком реально розширити його до рівня, який відповідає вашим потребам. Визначення позитивного чи негативного забарвлення тексту є основною метою аналізу настроїв. Давайте розглянемо кілька прикладів потенційних текстів для оцінки.

Позитивні:

- Цей музичний альбом неймовірний! Кожна пісня – це шедевр, який торкається душі.

- Цей мобільний додаток дуже зручний у використанні. Він значно спростив моє повсякденне планування.
- Цей парк чудовий для прогулянок. Чисте повітря, красиві ландшафти та спокійна атмосфера.

Негативні:

- Цей концерт був повною провальницею. Звук був жахливий, і артисти здавалися невідготовленими.
- Цей велосипед ламається через кожні два тижні. Я витратив більше грошей на ремонт, ніж на сам велосипед.
- Цей готель був дуже брудний і занедбаний. Кімнати були в жахливому стані, і я не почувався там комфортно.

Прості приклади:

- Цей кафе має найкращий капучіно у місті.
- Цей блог дуже інформативний та корисний.
- Ці кросівки дуже зручні та стильні.

Складні приклади:

- Ця комп'ютерна гра має захоплюючий геймплей, але графіка залишає бажати кращого.
- Ця вистава була досить гарною, але головна роль була виконана не надто переконливо.
- Цей смартфон має чудову камеру, але його батарея швидко розряджається.

У складних прикладах оцінювання стає більш контекстуальним і залежить від різних факторів. Однак, за допомогою сентимент аналізу можна визначити загальний настрій або позитивність/негативність кожного речення.

1.3.1 Методи класифікації тональності та класифікація текстів

Бінарна шкала

Тут ми говоримо про умови, де можуть відбуватися «позитивні» та «негативні» речі. Такий спосіб задати тон не завжди ефективний, тому що слова можуть заплутати. Тепер, коли вони вже досить дорослі, моделі здатні розуміти контекст повідомлень і іноді їх іронію.

Терні та Панг є двома першими дослідниками в цій галузі, які використовують різні методи для визначення полярності рецензій на продукти та фільми відповідно.[6]

Багатосмугова шкала

Панг і Снайдер [6] показали, що полярність можна класифікувати за допомогою багатосмугових шкал. Під цим ми маємо на увазі додавання 4-бальної системи оцінювання до категоризації. Рейтинг часто міститься в оглядах фільмів.

Системи шкалювання

Також поширеною є оцінка будь-чого за шкалою -10/10, -5/5 тощо. Ці методи порівнюють слова чи об'єкти з попередньо визначеними значеннями після першого використання комп'ютера для аналізу природної мови.

Зараз технології розвиваються настільки швидко, що комп'ютери можуть самостійно аналізувати тексти за допомогою штучного інтелекту та його алгоритмів. Ідея впливає з граматичних зв'язків між словами з урахуванням контексту[6].

Існують дві категорії методів аналізу тональності текстів, а саме такі способи:

- ручний
- автоматизований

Кінцева точність результатів означає ступінь варіації між різними техніками. Методи машинного навчання, які використовують інструменти автоматичної обробки природної мови, можуть обробляти великі обсяги даних.

В програмуванні існують такі види методів, що можуть бути застосовані до обробки природної мови, в нашому випадку до сентимент аналізу:

1. Методи, засновані на правилах і словниках

2. Машинне навчання з вчителем
3. Машинне навчання без вчителя
4. Метод, заснований на теоретико-графових моделях

Методи, засновані на правилах і словниках

Основою цього підходу є використання попередньо написаних правил і словників. Потім словникові концепції порівнюються за допомогою шкал, які базуються на тому, скільки слів включено в позитивний чи негативний текст.

Алгоритм методу побудований на основі словників і правил.

1. Тонова відповідність кожному слову оцінки
2. Підсумовування кількості позитивних і негативних слів.
3. Висловіть судження про весь текст.

Розвиток словникового запасу, який є складним і трудомістким, є лише одним із багатьох недоліків цього підходу. Для створення словника з неймовірно точними результатами вкрай важливо враховувати різні контексти, у яких використовуються конкретні поняття. Щоб гарантувати неймовірно потужні результати, цю стратегію слід ретельно застосовувати протягом тривалого періоду часу.

Машинне навчання з вчителем

Наразі найпоширенішим методом є машинне навчання з репетиторами. завдяки тому, що ці методи дають якісний результат і не займають багато часу та фізичної праці.

Алгоритм цього методу виглядає так:

1. Збір документів, на основі яких буде відбуватись навчання;
2. Дані розкладаються по ознакам, за якими буде відбуватись дослідження;
3. Зазначення тональності даних;
4. Вибір методу та алгоритму для навчання;
5. Застосування готової моделі.

НАВЧАННЯ З ВЧИТЕЛЕМ



Рисунок 1.5 - Навчання з вчителем (7)

Машинне навчання без вчителя

Цей метод полягає в тому, що комп'ютер дозволяє автоматично класифікувати об'єкти, мову чи інші пов'язані завдання. Початковий крок передбачає ідентифікацію термінів, які часто зустрічаються в одному тексті та рідко в іншому, у певному наборі текстів. У нашій вибірці ці терміни часто мають найвищу вагу. Визначення фраз і присвоєння їм тону дозволяє визначити інформацію щодо загального тону тексту. На практиці ці методи зазвичай застосовуються для підготовки попередніх даних.

Такі великі корпорації, як Google, також використовують текстову розмітку як маркетинговий інструмент. Наприклад, для підтвердження особи часто потрібна авторизація anticaptcha; вибравши відповідні зображення, можна як пройти ботоперевірку, так і навчити нейронні мережі точно ідентифікувати об'єкти. Ці алгоритми використовуються для таких завдань, як пошук правил, зменшення розмірності та кластеризація [7].

НАВЧАННЯ БЕЗ ВЧИТЕЛЯ



Рисунок 1.6 - Навчання без вчителя (7)

Метод, заснований на теоретико-графових моделях

Головною відмінністю цього методу є припущення, що не всі слова мають однакову тональність. Слова можуть мати різну вагу, що означає, що вони можуть мати більший вплив на загальний тон тексту.

При використанні цього методу аналіз тональності розбивається на кілька етапів:[6]

1. побудова графа на основі досліджуваного тексту;
2. ранжування його вершин;
3. класифікація знайдених слів;
4. обчислення результату.

Слова класифікуються за допомогою тонального словника, де кожне слово отримує оцінку, наприклад «позитивний», «негативний» або «нейтральний». Кінцева оцінка є сумою двох оцінок — позитивного та негативного кольору тексту. Підсумувавши та зваживши всі позитивні фрази в тексті, ви можете обчислити відсоток позитивності. Подібно до того, як письмо може мати погані конотації.

Треба визначити співвідношення цих елементів за допомогою такої формули, щоб визначити загальне судження про тональність тексту: Формула: $T = P/N$, де P

представляє оцінку позитивних елементів тексту, а N представляє негативні елементи тексту[6].

Дуже сприятливо, якщо він значно перевищує 1. Для текстів негативної тональності вірно і зворотне[6].

1.4 Порівняльна характеристика та формули розрахунків точності

Таблиця 1.1 - Порівняльна характеристика методів аналізу тональності тексту

Метод	Метод Методи, засновані на правилах і словниках	Машинне навчання з вчителем	Машинне навчання без вчителя	Метод, засновани й на теоретико -графових моделях
Плюси	Висока точність результатів	Висока точність, ефективне та швидке навчання	Не потребує завчасно підготовлено го пакету даних	Висока точність результатів
Мінуси	Велика трудомісткіс ть, висока затратність	Потребує завчасно підготовлено го пакету даних	Низька точність результатів	Складний у реалізації

Основними критеріями якості роботи будь якого алгоритму в програмуванні є точність та повнота результатів. Повнота обчислюється за наступною формулою:

$$R = \text{correctly extracted opinions} / \text{total number of opinions}$$

де *correctly extracted opinions* — правильно розпізнані думки, *total number of opinions* — загальна кількість думок (як знайдених системою, так і не знайдених)[6].

Точність результатів обчислюється за формулою:

$$P = \text{correctly extracted opinions} / \text{total number of opinions found by system}$$

де *correctly extracted opinions* — правильно розпізнані думки, *total number of opinions found by system* — загальна кількість думок знайдених системою[6].

1.5 Порівняльний аналіз алгоритмів CNN,BERT,RNN

Зараз існує кілька способів визначення тону тексту.

Одними з важливих методів є підходи на основі правил і словників, методи на основі теоретико-графічних моделей, контрольоване машинне навчання та неконтрольоване машинне навчання. Я прийняв рішення застосувати машинне навчання для вирішення проблеми тону тексту за допомогою вчителя. Це призвело до розробки та тестування сучасних алгоритмів, таких як BERT, CNN і RNN.

1.5.1 Двонаправлена нейронна мережа кодувальник - BERT

Google розробив BERT, техніку машинного навчання на основі Transformers, для попереднього навчання обробки природної мови. [8] Обраний підхід BERT має найкращі результати в дослідженнях обробки природної мови.

Логіка методу BERT

Контекст зв'язків слів у тексті вивчається машинним навчанням BERT за допомогою інструменту Transformer. Transformer використовує два механізми: кодер, який читає текст, і декодер, який робить передбачення для завдання. Оскільки метою методу BERT є створення мовної моделі, для цього потрібен лише механізм кодування.

Основним технологічним прогресом BERT є програма моделювання голосу популярної моделі навчання двостороннього трансформатора. Ця спроба досліджує обидва, на відміну від попередніх, які зосереджувалися на змішаному навчанні зліва направо або зліва направо або на послідовності тексту зліва направо. Дослідження показують, що двонаправлені мовні моделі краще сприймають контекст і потік мовлення, ніж однонаправлені мовні моделі [9].

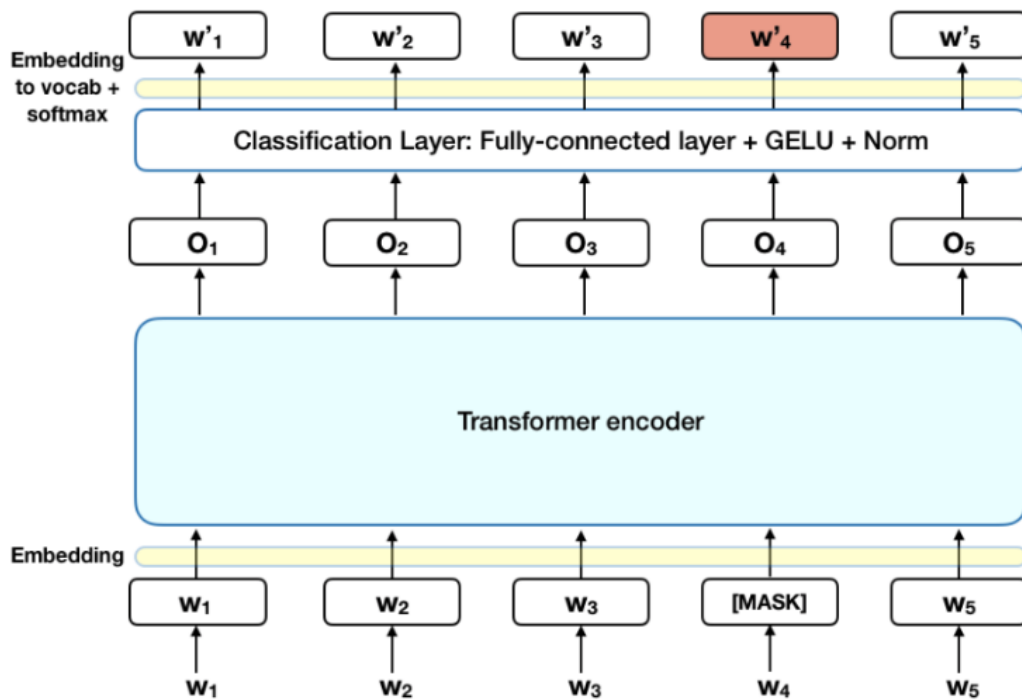


Рисунок 1.7 - Високорівневий опис кодера Transformer(9)

Вхідні дані складаються з набору токенів, які спочатку вбудовуються у вектори, а потім обробляються нейронною мережею. Результатом є набір векторів розміру H , що представляють кожен маркер у вхідних даних з однаковим індексом[9].

За задумом визначення контексту призводить до деяких проблем. Щоб вирішити цей конфлікт, модель BERT застосовує методи прогнозування наступного речення (NSP) і LM в масках (MLM).

Прогнозування наступного речення у послідовності (NSP)

Використовуючи пари варіацій як вхідні дані, розробка мовної моделі BERT вчиться оцінювати ймовірність того, що друге речення в отриманій парі є наступною фразою в тексті. Під час навчання вхідні пари поділяються на дві групи: ті, де друге речення є навімання вибраним реченням із корпусу, і ті, де друге речення є наступним у вихідному тексті [9].

Щоб модель могла правильно навчитись розрізняти два речення під час навчання, необхідно виконати наступні дії[9]:

1. На початку першого речення вставляється маркер [CLS], а в кінці кожного речення - маркер [SEP].
2. Вкладання речень за своїм поняттям схожі на вкладання символів із словниковим запасом 2.
3. Позиційне вбудовування додається до кожного маркера, щоб вказати його положення в послідовності.

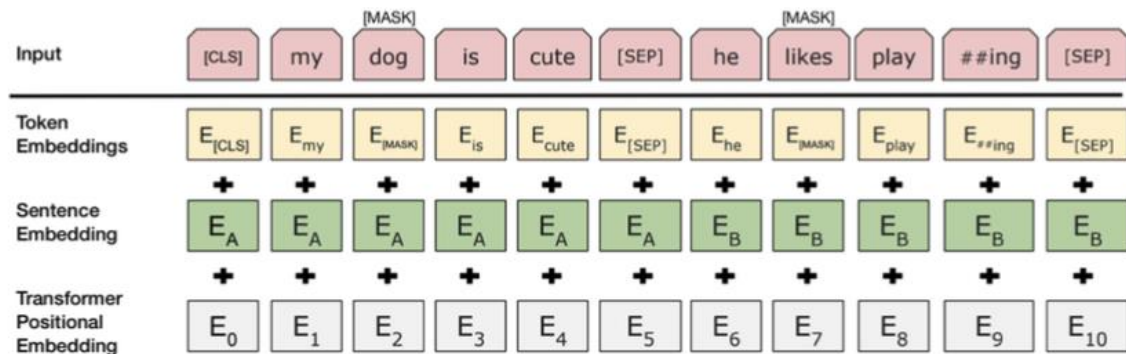


Рисунок 1.8 - Візуалізація попередньої обробки даних (9)

Щоб передбачити, чи друге речення справді пов'язане з першим, виконуються такі дії[9]:

1. Послідовність проходить через трансформатор
2. Вихід лексеми [CLS] перетворюється у вектор розміром 2×1 , використовуючи простий шар класифікації (вивчені матриці ваг та упереджень).
3. Обчислення ймовірності IsNextSequence за допомогою softmax.

Під час створення та вдосконалення моделі BERT усі процедури виконуються одночасно, щоб мінімізувати спільну функцію втрат двох підходів.

Група, яка розробила моделі BERT, застосувала техніку, яка передбачає, що параметри повинні залишатися початковими, щоб отримати такі високі результати.

1.5.2 Згорткова нейронна мережа

Згорткові нейронні мережі — це один із видів автоматизованих нейронних мереж, який часто використовується для обробки зображень. Незважаючи на це, емоційний контекст тексту також з'ясовується за допомогою мережевих даних.

Як влаштована мережа

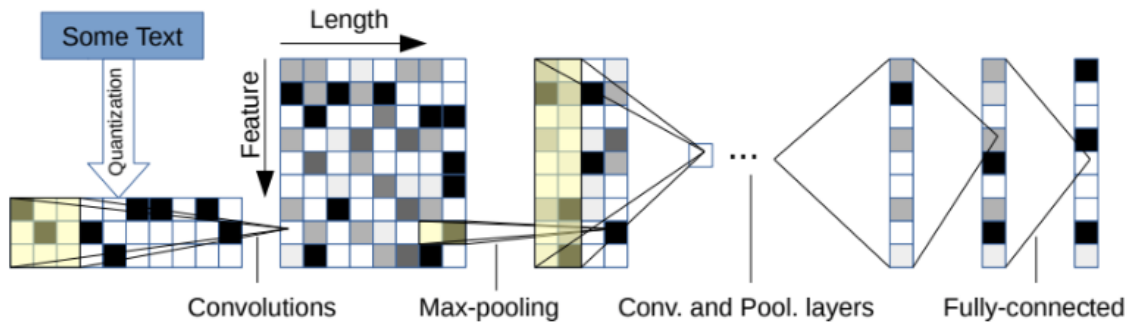


Рисунок 1.9 - Візуалізація алгоритму CNN з текстом(11)

Згорткова мережа може використовуватися лише з матрицею; ці мережі не працюють без нього. Для цього використовується оболонка — техніка відображення точки в багатовимірному просторі об'єкта — в даному випадку слова.

Word-to-Vec

Все, що описує word2vec, це відображення точки на об'єкт, у цьому випадку слово. Ось як працює програма[10]:

Word2vec використовується для генерації координат слова шляхом відображення вектора на кожне слово з величезного текстового корпусу. Спочатку він створює корпусний словник, а потім створює векторне представлення слів, «навчаючись» із вхідних текстів.

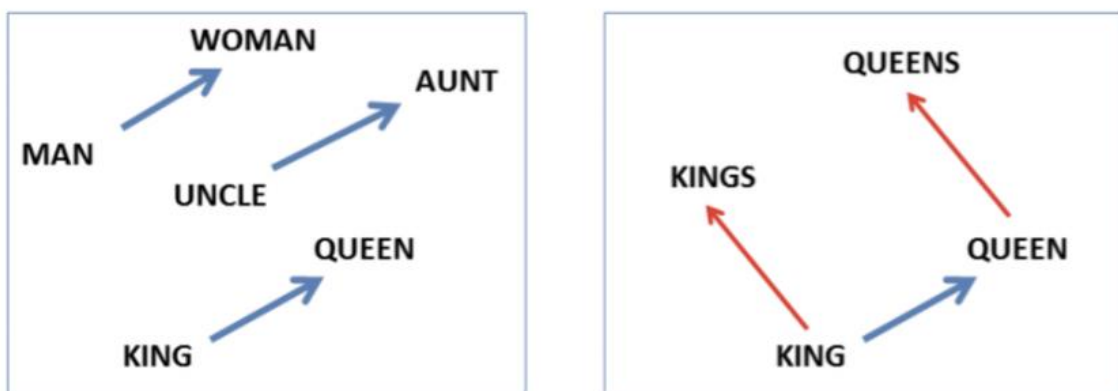


Рисунок 1.10 - Word2Vec(11)

Логіка алгоритму, що використовується CNN

Застосування вищезгаданої техніки word2Vec дасть матрицю, яку використовує мережа. Матриця має три виміри: канали, висоту та ширину. Вхідна матриця спочатку обробляється за допомогою шарів згортки. Карта ознак, яку ми отримуємо на виході кожного фільтра, потім обробляється за допомогою шару підвибірки з указаною функцією завантаження, таким чином ущільнюючи карту функцій. Це дозволяє нам отримати інформацію про згортку, яку ми можемо використовувати для класифікації, незалежно від того, де вона з'являється в тексті.

На основі згенерованих карт необхідно створити один векторний об'єкт. Потім обчислення міток класу виконуються на вихідному рівні нейронної мережі та на вхідному рівні прихованого повнозв'язного рівня.

1.5.3 Рекурентна нейронна мережа

Рекурентні нейронні мережі є підкласом штучних нейронних мереж, які складаються з вузлових зв'язків, які утворюють орієнтований у часі граф [12]. Ці нейронні мережі мають два входи для активації на додаток до вхідного та вихідного зворотного зв'язку. Цей вихід повертається в той самий блок з новими входами, коли всі входи були використані.

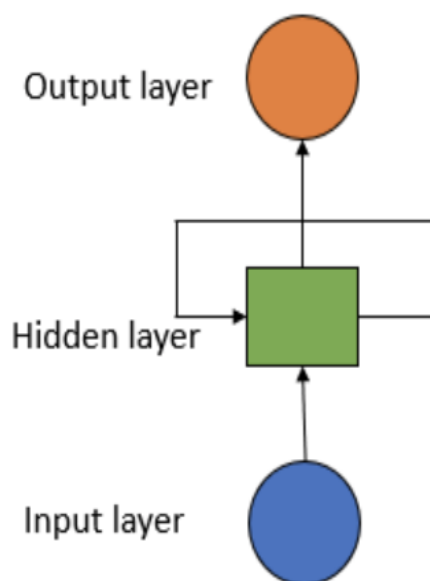


Рисунок 1.11 - Візуалізація шарів (13)

Побудова рекурентних нейронних мереж включає три кроки. Прихований шар має пройти текст, перш ніж він зможе передбачити фразу. На другому кроці функція втрат використовується для порівняння прогнозу та правильного значення. Ефективність нашої моделі відображається функцією втрат. Останнім кроком є визначення градієнта для кожної точки або вузла. Градієнт, що дорівнює значенню [13], змінює вагу мережі в кожному місці.

Рекурентні нейронні мережі часто використовуються з послідовними даними. Використання шарів надає моделі короткочасну пам'ять. Цей метод забезпечує точність результатів. Ваги, призначені історичним даним, визначають, як довго вони зберігатимуться, а не вказаний період часу. RNN використовується для аналізу настроїв, мовлення тегами, послідовностей тегів тощо[13].

Шари рекурентної нейронної мережі

Одним із важливих етапів є вбудовування згорткової нейронної мережі. Першим параметром є кількість слів. По-друге, кількість векторів, які складають це вкладення. Кожне слово в корпусі має відповідний розмір у тексті.

Другий рівень, або LSTM, у цих типах нейронних мереж є найважливішим елементом у створенні рекурентних нейронних мереж. Цей шар допомагає вирішити проблему зникнення градієнта та надає можливість передбачити наступне слово за допомогою короткочасної пам'яті.

Зникаючі градієнти

Для зміни ваги в кожній позиції використовується градієнт. Інтенсивність градієнта визначає коригування.

Якби значення градієнта попереднього шару було меншим, таким же було б і значення градієнта в цьому вузлі, і навпаки [13]. У випадку, якщо градієнти невеликі, тренування не приносять користі, оскільки це означає, що вага недостатньо контролюється весь час. У цьому випадку модель не зможе зрозуміти контекст слів.

Рішенням цього питання є використання LSTM. У зворотному поширенні довготривала пам'ять буде контролювати потік даних[13]. Втрата ваги зупиняється за допомогою вбудованого двигуна LSTM. Таким чином, модель набуває більшої

глибини та підвищеної ефективності. Двигун точно передбачає послідовність, оскільки передається лише відповідна інформація, а зайва видаляється на вході в структуру.

Вихідний шар

«Сигмовид» — це функція, яка використовується на першому рівні. Однією з переваг функції активації є те, що вона використовує лише значення, необхідне для передбачення. Проводиться груповий інструктаж. Формуючи групи, комп'ютер може виконати завдання за значно менший проміжок часу.

Таблиця 1.2 - Порівняльна характеристика методів

	BERT	CNN	RNN
Втрата NSP	-	35%	31%
Точність	93%	81%	86%

Обраний алгоритм для тестування за допомогою телеграм-бота

В результаті порівняння та дослідження різних типів методів та алгоритмів, мій вибір пав на попередньо навчену модель RoBERTa методу BERT, що буде застосовано у телеграм боті. Для підключення використовується модифікація **sentiment-roberta-large-english**.

Дана модель надає результат у одновимірному просторі ми отримуємо позитивні (1) або негативні (0) настрої. Модель виділяється тим, що була навчена на 15 видах текстів, тому чудово справляється з різними текстовими даними, різними контекстами тощо, а отже, результат значно вищий, порівняно з результатами інших методів.

Модель RoBERTa була запропонована в RoBERTa: Надійно оптимізований підхід до підготовки BERT від Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov. Вона заснований на моделі BERT від Google, випущеній у 2018 році.[14]

Модель базується на BERT та модифікує ключові параметри, що дає можливість навчати модель за допомогою великих міні-партій, що робить модель найбільш ефективною серед конкурентів.

Таблиця 1.3 - Параметри для попередньої підготовки RoBERTa - large[16]

Параметри	RoBERTa - large
Кількість шарів	24
Прихований розмір	1024
FFN внутрішній прихований розмір	4096
Головки уваги	16
Розмір головки уваги	64
Випадання	0.1
Увага, що випадає	0.1

Кроки підготовки	30k
Піковий рівень навчання	4e-4
Розмір партії	8k
Зниження ваги	0.01
Максимальний крок	500k
Зниження швидкості навчання	Лінійна
Adame ϵ	1e-6
Adam β_1	0.9
Adam β_2	0.98
Відсікання градієнта	0.0

Таблиця 1.4 - Параметри для налаштування моделі RoBERTa LARGE[16]

Гіпер параметри	RACE	SQuAD	GLUE
Швидкість навчання	1e-5	1.5e-5	{1e-5, 2e-5, 3e-5}
Розмір партії	16	48	{16,32}
Зниження ваги	0.1	-	-
Зниження швидкості навчання	Лінійна	Лінійна	Лінійна
Коефіцієнт підготовки	0.06	0.06	0.06
Максимальна кількість епох	4	2	10

1.5.4 Порівняння моделі RoBERTa з іншими модифікаціями методу BERT

Розглянемо результати досліджень та тестування попередньо навченої моделі RoBERTa за трьома різними стандартними еталонами: GLUE, SQuAD та RACE.

Таблиця 1.5 - Результати за оцінкою RACE [16]

Модель	Точність
BERT_large	72%
XLNet_large	81.7%
RoBERTa	83.2%

Таблиця 1.6 - Результати отримані за еталоном GLUE [16]

Модель	EM	F1
BERT_large	79%	81.8%
XLNet_large	86.1%	88.8%
RoBERTa	86.5%	89.4%

EM - показник що відображає відсоток, який точно відповідає будь-якій із основних правдивих відповідей , F1 - показник що відображає середнє значення між передбаченням та відповіддю на істинність.

Таблиця 1.7 - Результати отримані за еталоном SQuAD[16]

Модель	MNLI	SST
BERT_large	86.6%/-	93.2%
XLNet_large	89.8%/-	95.6%
RoBERTa	90.2%/90.2%	96.4%

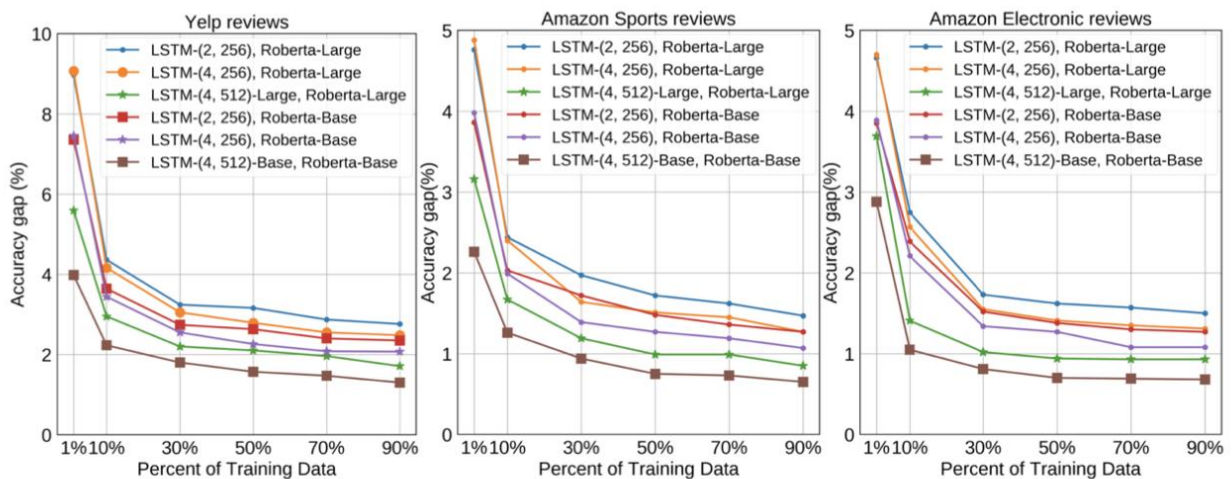


Рисунок 1.12 -Точність моделі RoBERTa, за використанням різних датасетів (15)

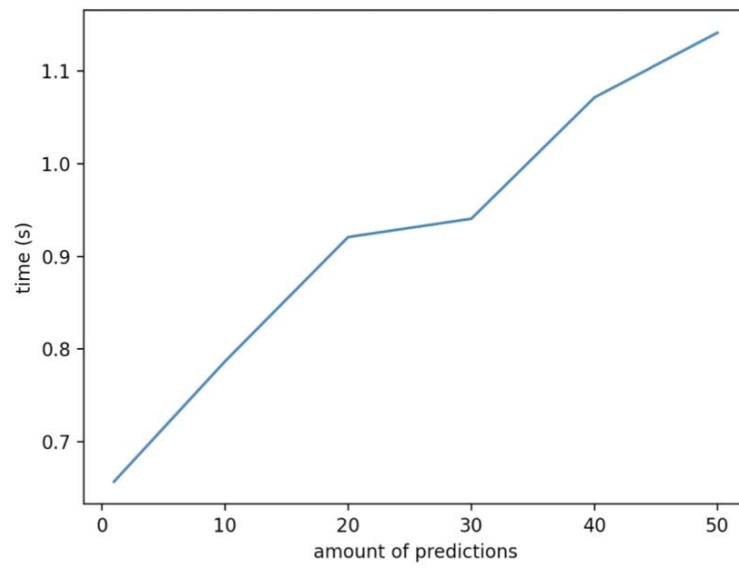


Рисунок 1.13 - Графік швидкості роботи моделі RoBERTa

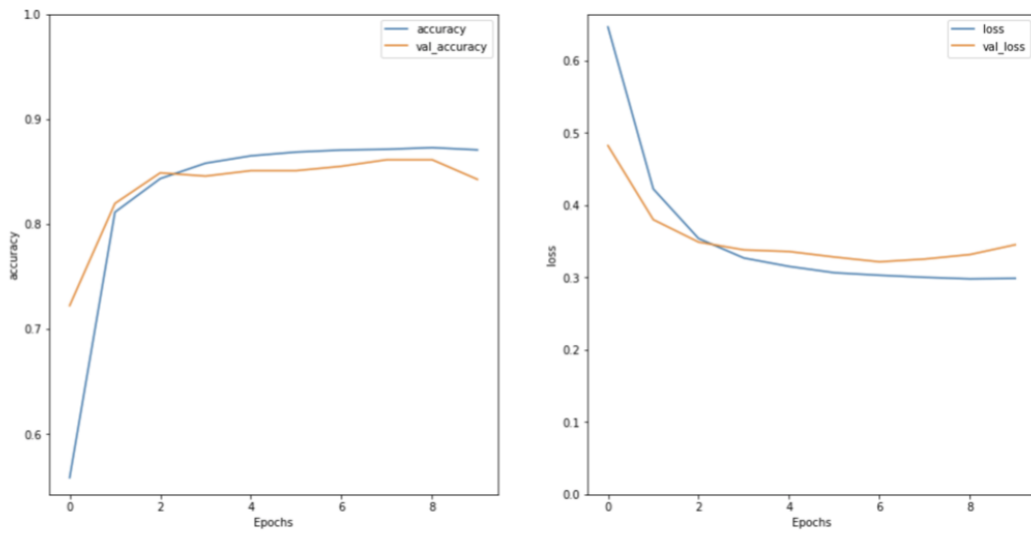


Рисунок 1.14 - Графік точності рекурентних нейронних мереж в залежності від кількості епох

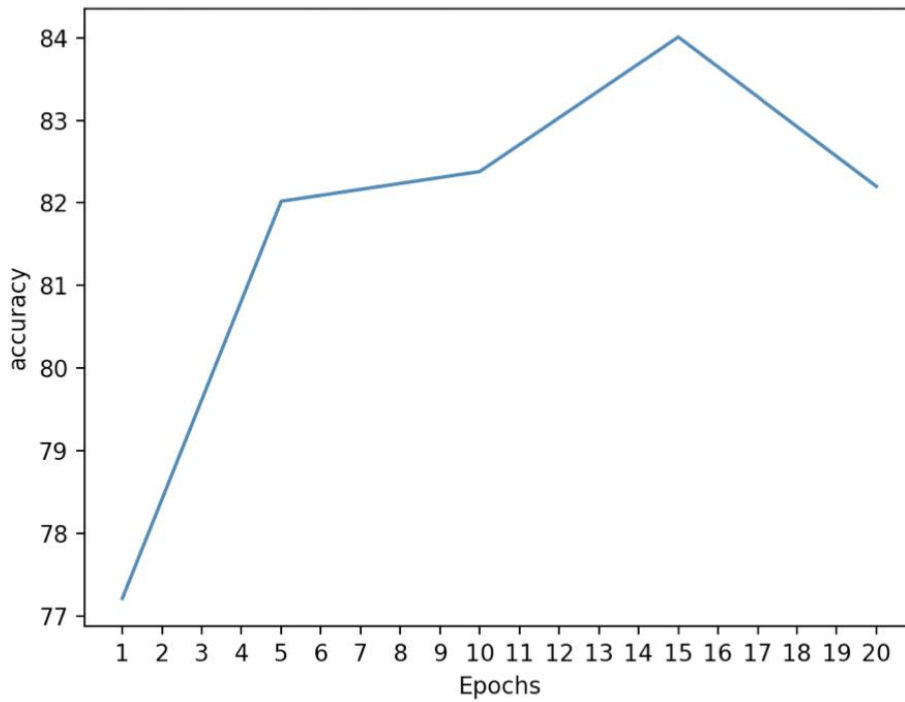


Рисунок 1.15 - Графік точності для алгоритму згорткових нейронних мереж в залежності від кількості епох

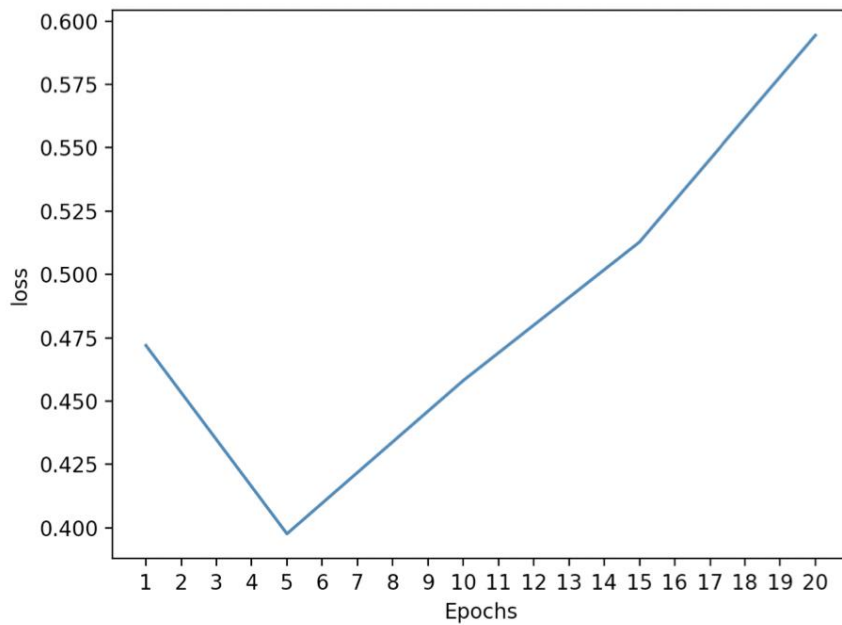


Рисунок 1.16 - Графік втрат NSP для алгоритму згорткових нейронних мереж в залежності від кількості епох

1.6 Постановка задачі

Аналітика та моніторинг соціальних медіа представляють великий інтерес для державних установ, лінгвістів, психологів, маркетологів і соціологів.[2]

Дослідження настроїв клієнтів часто використовують цей підхід. Штучний інтелект має ширше застосування, ніж, скажімо, опитування, до яких ми звикли, оскільки він може самостійно збирати та аналізувати важливі дані. Це також позбавляє потреби в більшій кількості дослідників для проведення досліджень.

Таким чином, мета цього дипломного проекту полягала в тому, щоб створити програму для зчеплення моделей, яка вибирається шляхом порівняння кількох моделей, які вирішують проблему аналізу настроїв.

Метою цього дослідження є порівняння різних методів аналізу настроїв, таких як згорткові нейронні мережі, рекурентні нейронні мережі та двонаправлені трансформатори Берта. Пов'язану модель, яка буде обрана в результаті порівняння, буде перевірено за допомогою інтегрованого бота Telegram. Англійські тексти підтримуються ботом Telegram.

Було сформовано такі завдання для виконання поставлених задач:

- Дослідження сфер, де використовується задача сентимент аналізу
- Негативні наслідки використання застарілих гаджетів
- Аналіз та порівняння основних алгоритмів, та обрання моделі для тестування
- Тестування
- Висновки

Висновки до розділу 1

Сучасний світ надає великого значення аналізу настроїв, оскільки його можна застосувати до широкого діапазону сценаріїв і сфер, таких як прогнозування фондового ринку, обслуговування клієнтів і маркетинг. Проблеми з обробкою природної мови можна вирішити різними способами. Усі вони досягають тієї самої мети, незважаючи на те, що мають різні механізми та різні рівні точності та швидкості обчислень.

Незважаючи на те, що тональний аналіз важливий у сучасному світі, не було створено жодного інструменту, який би дозволив будь-кому аналізувати текст у соціальній мережі Telegram для бізнесу чи особистих цілей.

Ця дисертація охоплює підходи машинного навчання для обробки тексту, згорткових і нейронних мереж, а також двонаправленого трансформатора BERT. За результатами аналізу буде обрано найбільш успішну попередньо навчену модель одного з розглянутих алгоритмів і буде активовано Telegram-бот з можливістю оцінки тональності повідомлень.

2 РЕАЛІЗАЦІЯ ЧАТ-БОТА

2.1 Використані технології

В даній роботі були використані такі технології:

- мова програмування - Python 3
- TelegramBotAPI - бібліотека для створення бота
- SublimeText 3 - редактор

Застосунок може відтворюватись на будь якій операційній системі, за рахунок кросплатформеності мови програмування.

2.2 Опис методів

Метод `analyze_sentiment`

Призначення: ця функція є основою здатності бота аналізувати настрої. Він приймає повідомлення користувача, аналізує його настрої за допомогою попередньо навченої моделі машинного навчання (у цьому випадку DistilBERT), а потім надсилає відповідь користувачеві.

Логіка: спочатку витягує текст із повідомлення користувача (`user_text = update.message.text`). Потім він використовує модель аналізу настроїв (`sentiment_model`), щоб оцінити настрої тексту. Результат включає позначку настрою (позитивну чи негативну) і оцінку достовірності. Потім функція створює рядок відповіді, що вказує на настрої та оцінку достовірності, який надсилається назад користувачеві.

Метод `start`

Призначення : ця функція служить початковою точкою взаємодії для нових користувачів. Коли користувач починає розмову з ботом (зазвичай, надсилаючи команду `/start`), ця функція надсилає вітальне повідомлення.

Логіка: метод просто відповідає користувачеві вітальним повідомленням та інструкціями щодо використання бота.

Метод `main`

Призначення : це точка входу сценарію. Він налаштовує бота Telegram, реєструє обробники (функції для обробки різних типів подій або повідомлень) і запускає бота. Логіка: функція спочатку створює Updater об'єкт із маркером бота. Об'єкт Updater відповідає за отримання нових оновлень з Telegram і доставку їх боту. Потім він отримує dispatcher від засобу оновлення, який використовується для додавання обробників. Функція start зареєстрована як обробник команд для /start команд, а analyze_sentiment функція зареєстрована як обробник повідомлень для текстових повідомлень. Нарешті бот починає запитувати оновлення.

Кожен метод відіграє певну роль, дозволяючи боту взаємодіяти з користувачами, аналізувати їхні повідомлення та відповідати відповідним чином. Загальна архітектура розроблена як модульна, що дозволяє кожній частині функціонувати незалежно, але разом з тим сприяти загальним можливостям бота.

2.3 Підключення моделі

Щоб підключити обрану мною модель, а саме sentiment-roberta-large-english необхідно модифікувати результати у потрібну форму, для відправки їх у чат.

Для цього треба виконати такі дії, як:

За допомогою методу оновлення вихідні дані моделі змінюються, створюючи новий словник зі списку словників, який містить інформацію, необхідну для перевірки моделі. Два ключі в новому словнику: «мітка», яка містить дані про емоційний зміст тексту, і «оцінка», яка містить значення від -1 до 1, які вказують, наскільки точна оцінка.

У роботі нам потрібен тільки ключ зі значенням “positive” або “negative” для відображення результату.

А “score” було використано для побудови графіків точності.

2.4 Опис програмного продукту

Telegram бот написаний мовою програмування python 3. Обрано модель методу Bert, а саме модель RoBerta Large **sentiment-roberta-large-english**[15], ця модель є однією із найбільш відпрацьованих в задачі природної мови.

Причиною високого результату, який відрізняє його від інших підходів, є те, що модель навчається та оцінюється за допомогою даних із 15 різних текстових джерел.

Таким чином, ця модель працює краще, ніж інші, які були навчені на одному типі тексту.

2.5 Моделювання програмного забезпечення

2.5.1 Діаграма використання

Діаграма використання (Use Case Diagram) в UML (Unified Modeling Language) — це важливий інструмент в області аналізу та проектування програмного забезпечення. Вона використовується для візуалізації функціональних вимог до системи, відображаючи взаємодії між користувачами (або "акторами") і випадками використання системи.



Рисунок 2.1 - Діаграма використання

На діаграмі використання зображено такі можливості користувача, як:

1. Класифікація тексту, що складається з кількох речень - тобто це означає, що модель дозволяє тестувати тексти різної довжини.
2. Введення тексту - користувач може ввести будь який текст.
3. Класифікація тексту, що складається з одного речення - так само як і у випадку текстів із кількох речень.
4. Отримання результатів - після введення тексту та обробки, користувач отримує результат.

2.5.2 BPMN

Для подальшої розробки було розроблено схему BPMN (Business Process Model and Notation) – модель та позначення бізнес-процесів.

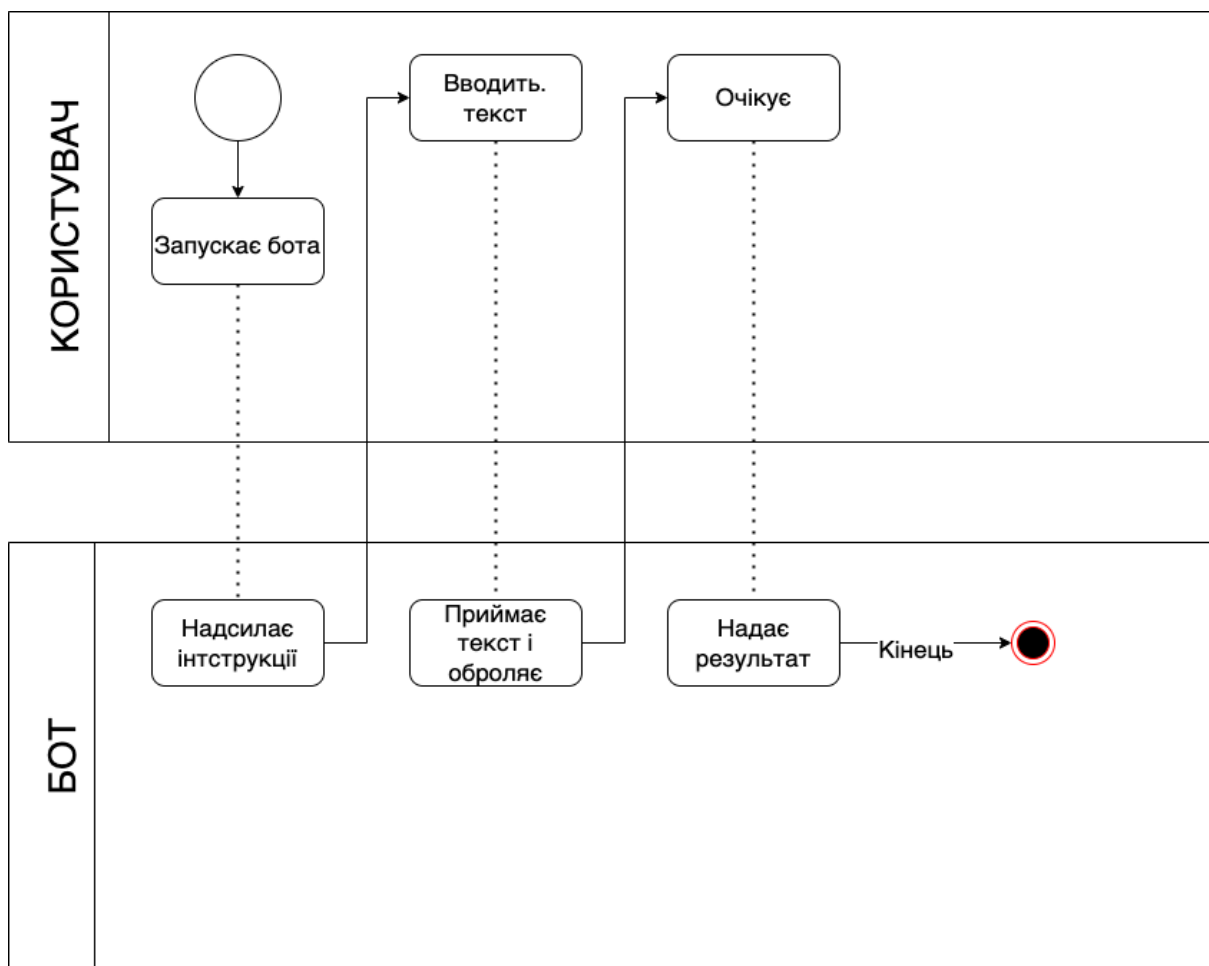


Рисунок 2.2 - Схема роботи бота

Опис роботи бота:

- користувач запускає бота
- бот надає інструкції
- користувач вводить текст і очікує
- бот обробляє текст
- бот надає результат

Так як запуск бота відбувається через командну строку, то використання IDE при розгортанні не обов'язкове.

2.5.3 Діаграма активності

Діаграма активності - це один із типів діаграм, що використовуються в області моделювання бізнес-процесів та системного аналізу. Вона є частиною нотації UML (Unified Modeling Language) і використовується для візуалізації робочих процесів або поведінки системи.

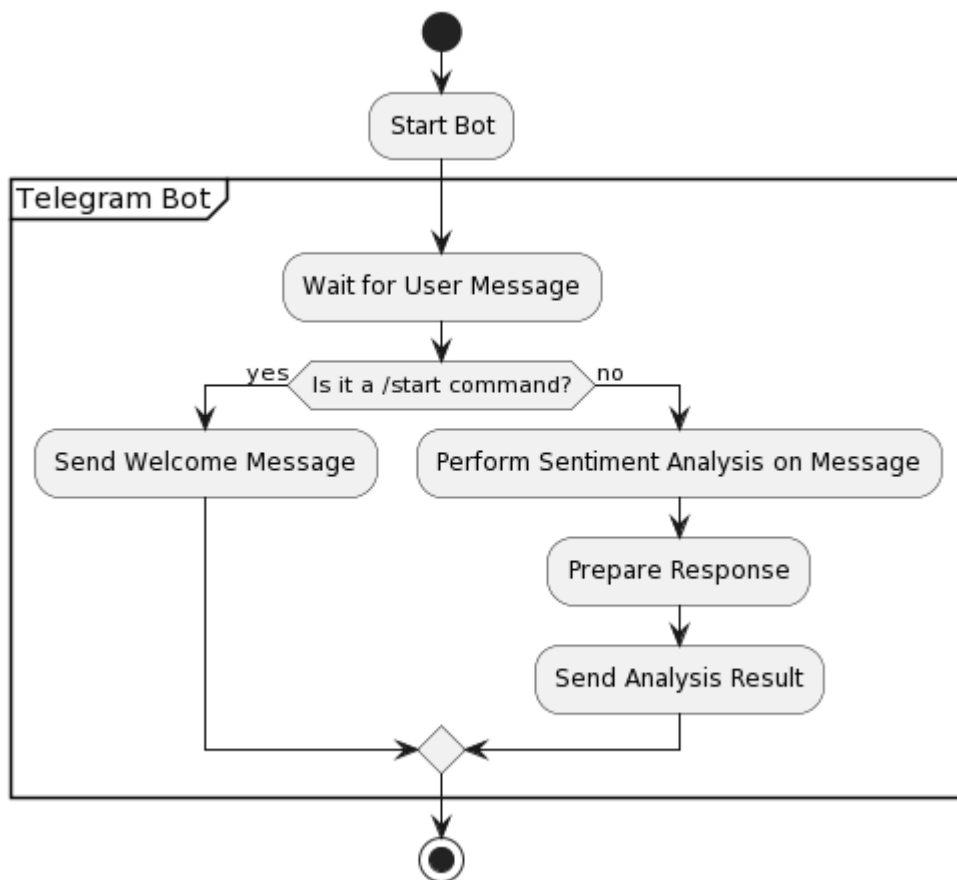


Рисунок 2.3 - Діаграма активності

Процес:

1. Процес починається з «Запустити бота».
2. Потім бот переходить у стан, коли очікує повідомлення користувача.
3. Існує точка прийняття рішення, яка перевіряє, чи є отримане повідомлення командою /start.
4. Якщо так, бот надсилає вітальне повідомлення.
5. Якщо ні, бот виконує аналіз настрою повідомлення, готує відповідь і надсилає результат аналізу назад користувачеві.
6. Потім процес завершується.

Висновки до розділу 2

Для реалізації практичної частини було використано мову Python 3, метод Bert та бібліотеку TelegramBotApi. Для відображення результатів , було зроблено перетворення за допомогою методу update. У даному розділі здійснено проектування телеграм бота та описано його структуру та архітектуру.

3 ТЕСТУВАННЯ ТЕЛЕГРАМ БОТА

3.1 Опис методу тестування

Для тестування було обрано функціональне тестування.

Таблиця 3.1 – Мінімальні системи вимоги для запуску застосунку

Операційна система	Windows 7 , macOS 10.12, Ubuntu 16.04
CPU (процесор)	Intel Core 3 та вище
RAM (оперативна пам'ять)	2 GB
Об'єм пам'яті на диску	2 GB

Таблиця 3.2 – Перевірка функціоналу запуску бота

Мета тесту	Перевірка функціоналу початку роботи бота
Назва сцени, на якій проводиться випробування	Початковий екран
Схема проведення тесту	Користувач натискає команду "Start"
Очікуваний результат	Бот надсилає повідомлення з проханням ввести текст
Результат випробування	Результат збігається з очікуваним

Таблиця 3.3 – Перевірка введення тексту для класифікації

Мета тесту	Перевірка введення тексту для класифікації
Назва сцени, на якій проводиться випробування	Початковий екран
Схема проведення тесту	Користувач вводить текст
Очікуваний результат	Користувач вводить текст і він приймається на обробку ботом
Результат випробування	Результат збігається з очікуваним

Таблиця 3.4 – Перевірка проведення класифікації

Мета тесту	Перевірка проведення класифікації
Назва сцени, на якій проводиться випробування	Початковий екран
Схема проведення тесту	Бот обробляє текст і надсилає повідомлення з результатом
Очікуваний результат	Бот надсилає результат
Результат випробування	Результат збігається з очікуваним

Варіанти дій користувача в телеграм боті та їх опис наведено у таблицях

Таблиця 3.5 – Опис варіанту використання UsC1

Назва	Запуск бота
Опис	Перехід по посиланню чи qr-code до бота та його запус
Дійова особа	Користувач
Передумови	Користувач перейшов по посиланню
Постумови	Користувач запустив бота
Основний сценарій	Користувач запускає бота

Таблиця 3.6 – Опис варіанту використання UsC2

Назва	Отримати інструкції з користування
Опис	Бот повинен надіслати повідомлення про те, що користувач має ввести повідомлення англійською мовою для класифікації
Дійова особа	Користувач
Передумови	Користувач запустив бота
Постумови	Користувач отримав повідомлення від бота
Основний сценарій	Користувач отримав повідомлення від бота

Таблиця 3.7 – Опис варіанту використання UsC3

Назва	Введення тексту
Опис	Користувач вводить текст який хоче класифікувати
Дійова особа	Користувач
Передумови	Користувач отримав повідомлення від бота
Постумови	Користувач ввів текст
Основний сценарій	Користувач ввів текст

Таблиця 3.8 – Опис варіанту використання UsC4

Назва	Отримання результатів
Опис	Користувач отримує результати класифікації після обробки тексту
Дійова особа	Користувач
Передумови	Користувач ввів текст
Постумови	Користувач отримує результати класифікації після обробки тексту
Основний сценарій	Користувач отримує результати класифікації після обробки тексту

Нижче описано функціональні вимоги до програмного забезпечення:

Таблиця 3.9 – Функціональна вимога F1

Назва	Запустити бота
Опис	Можливість запустити бота нажавши відповідну команду

Таблиця 3.10 – Функціональна вимога F-2

Назва	Введення тексту
Опис	Можливість ввести текст для обробки моделлю

Таблиця 3.11 – Функціональна вимога F-3

Назва	Обробка моделлю
Опис	Можливість обробки тексту моделлю

Таблиця 3.12 – Функціональна вимога F-4

Назва	Отримання результатів
Опис	Можливість отримати результат про проведену класифікацію

Матриця трасування на базі варіантів використання та функціональних вимог виглядає наступним чином:

Таблиця 3.13 – Матриця трасування вимог

	UsC1	UsC2	UsC3	UsC4
F-1	+		+	
F-2	+			+
F-3	+	+	+	
F-4		+		+

Згідно з матрицею трасування, основні функціональні вимоги повністю покриваються представленими варіантами використання.

3.2 Приклад використання

Як і зі всіма телеграм ботами, для початку роботи треба ввести команду /start.

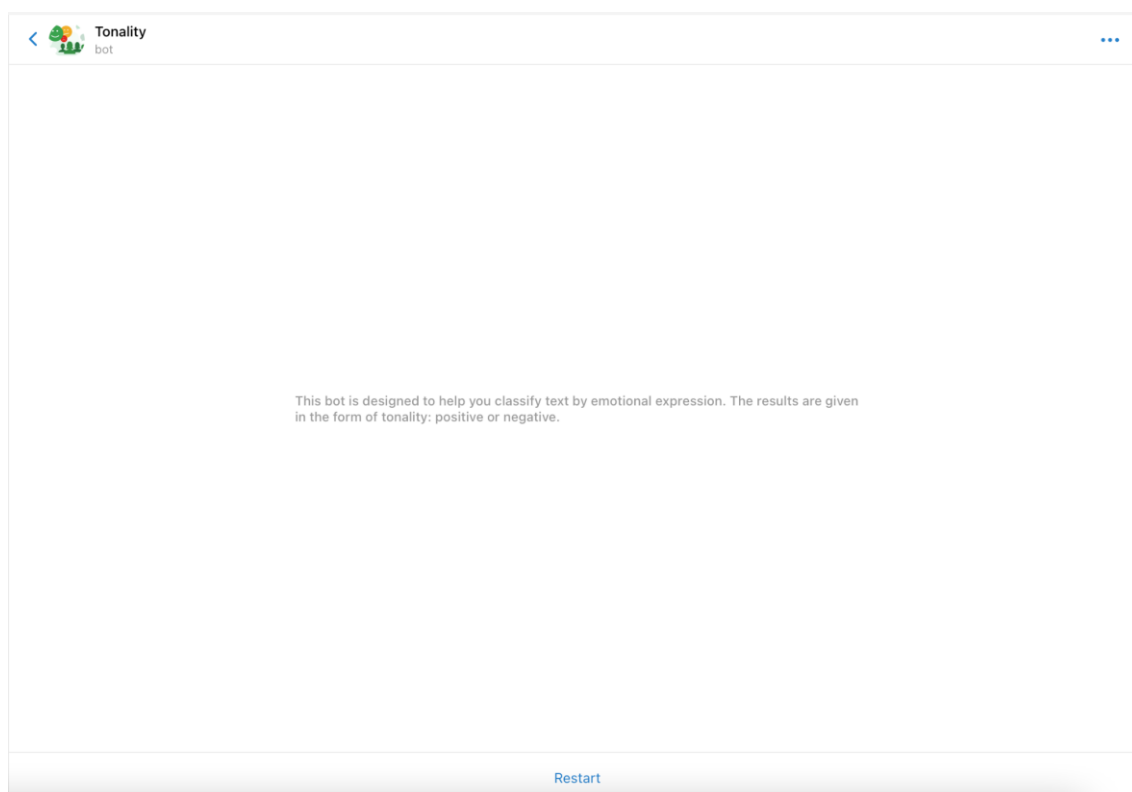


Рисунок - 3.1 Початок роботи

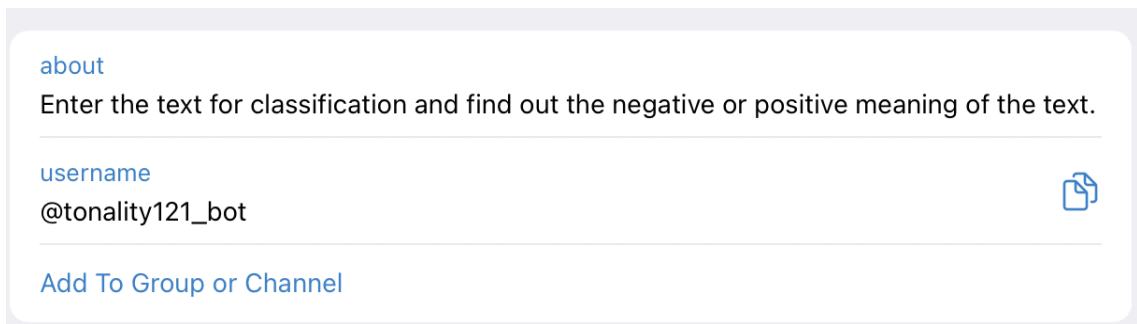


Рисунок - 3.2 Опис

На початку бот надішле в чат повідомлення з проханням надіслати текст для проведення класифікації на емоційне забарвлення.

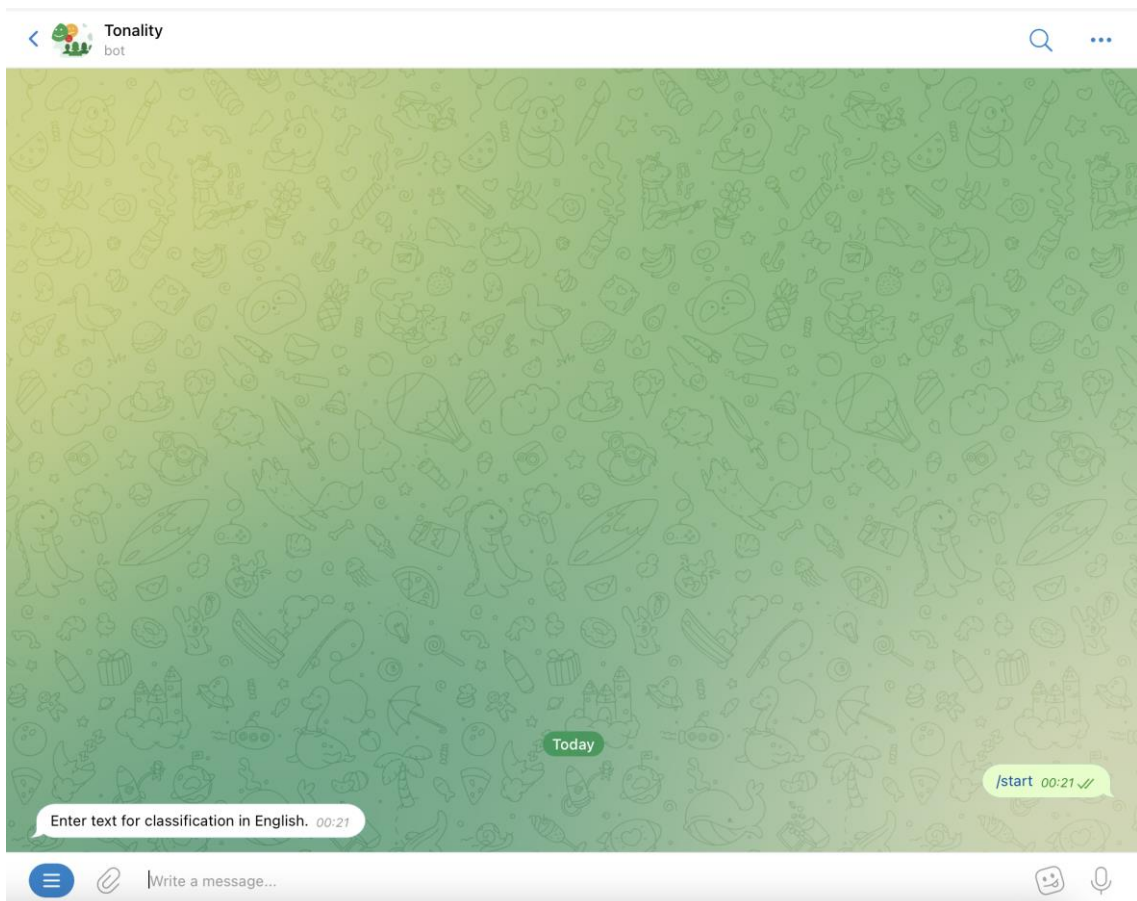


Рисунок - 3.3 Робота телеграм бота

Після введення тексту, бот обробить введені дані та надасть повідомлення з результатом.

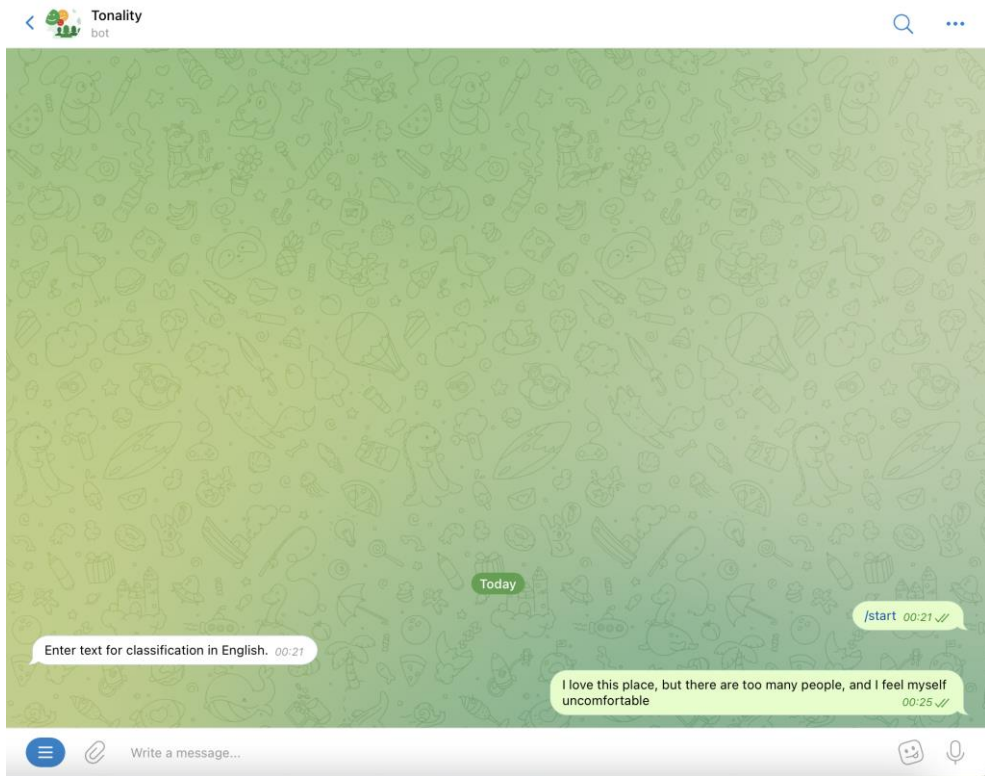


Рисунок - 3.4 Приклад введення тексту

Через деякий час телеграм бот надсилає нам повідомлення в якому йдеться про результат класифікації.

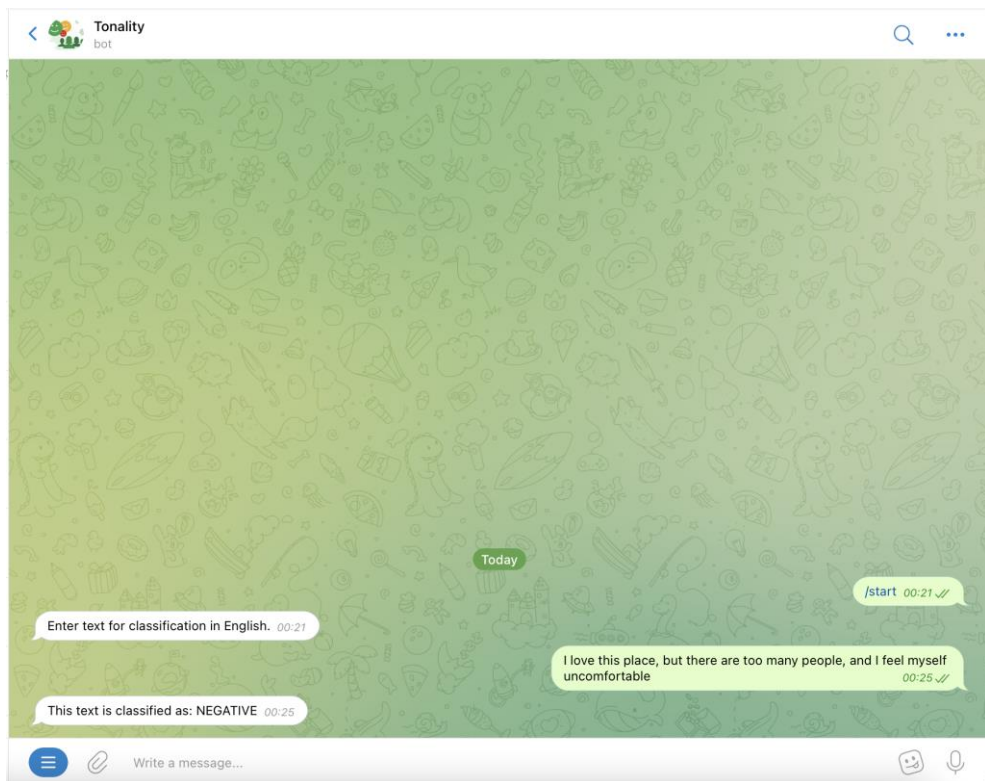


Рисунок - 3.5 Результати класифікації

Результат класифікації надсилається ботом у вигляді повідомлення.
Спробуємо розглянути ще кілька прикладів.

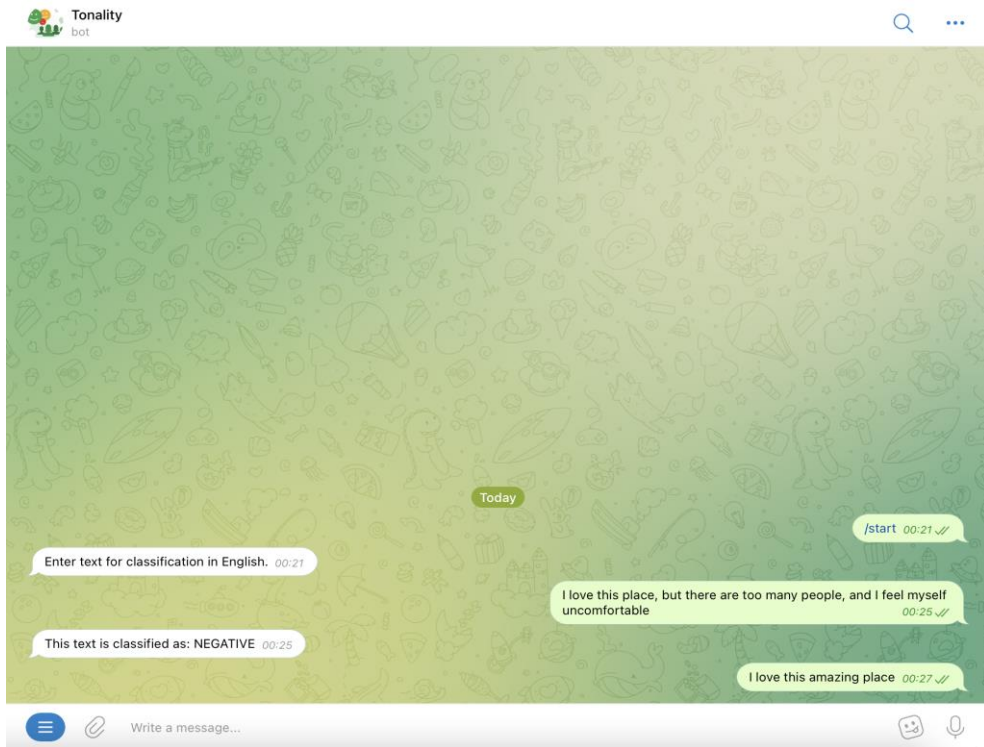


Рисунок - 3.6 Приклад введення тексту

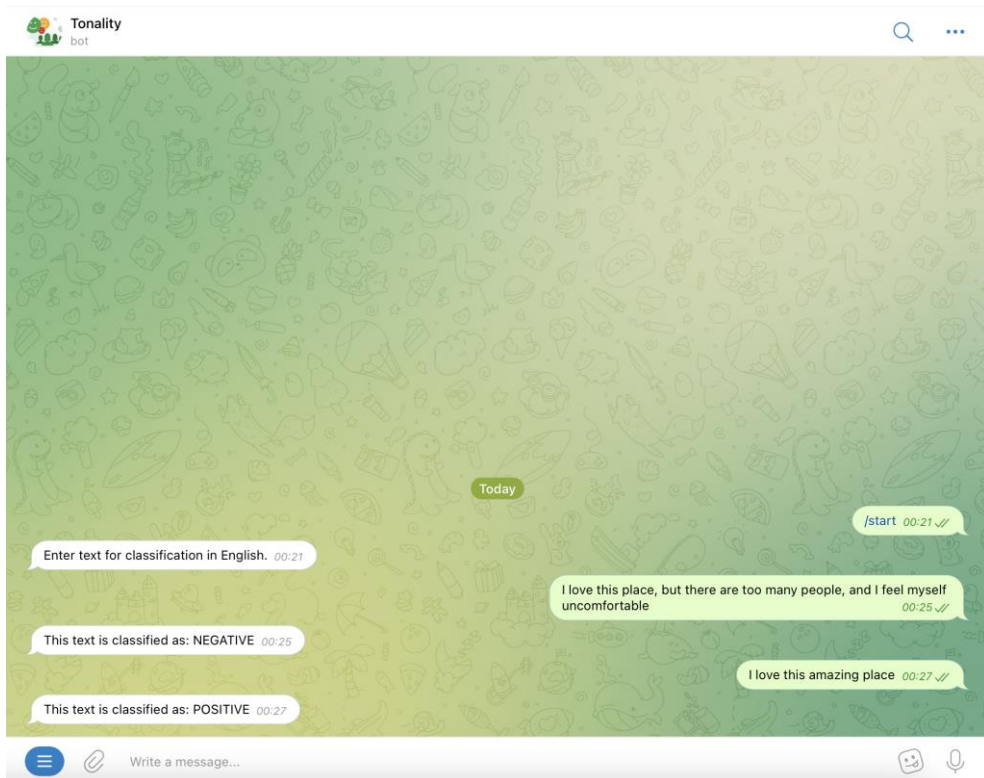


Рисунок - 3.7 Результати класифікації

Як бачимо бот працює коректно. Тепер спробуємо проаналізувати текст, що складається з декількох речень.

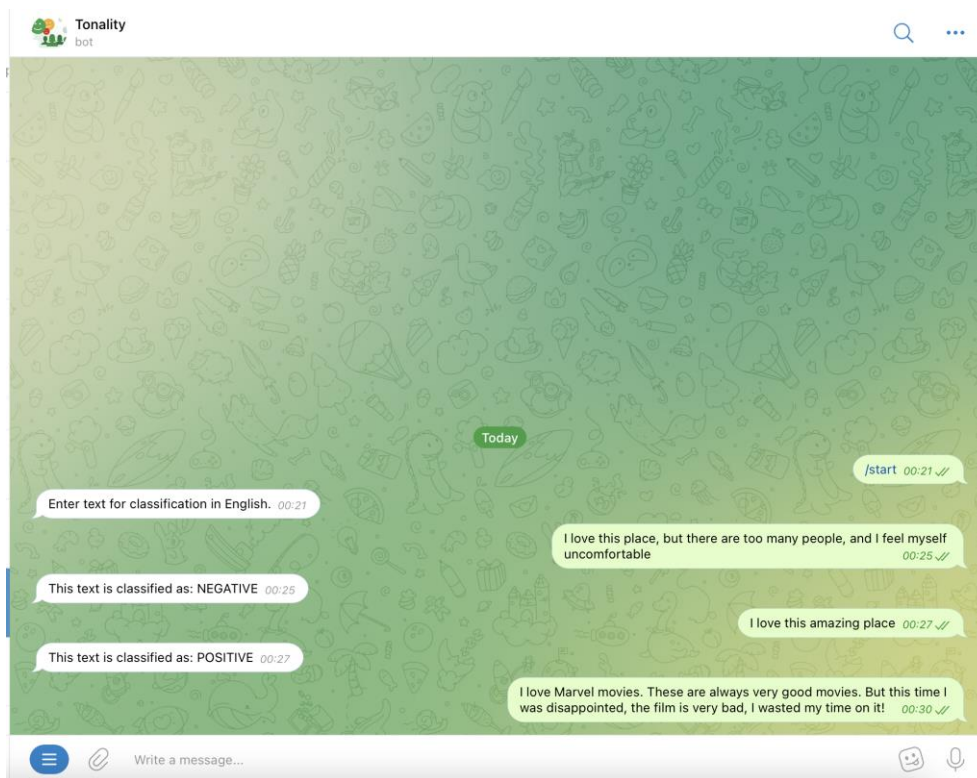


Рисунок - 3.8 Приклад введення тексту з декількома реченнями

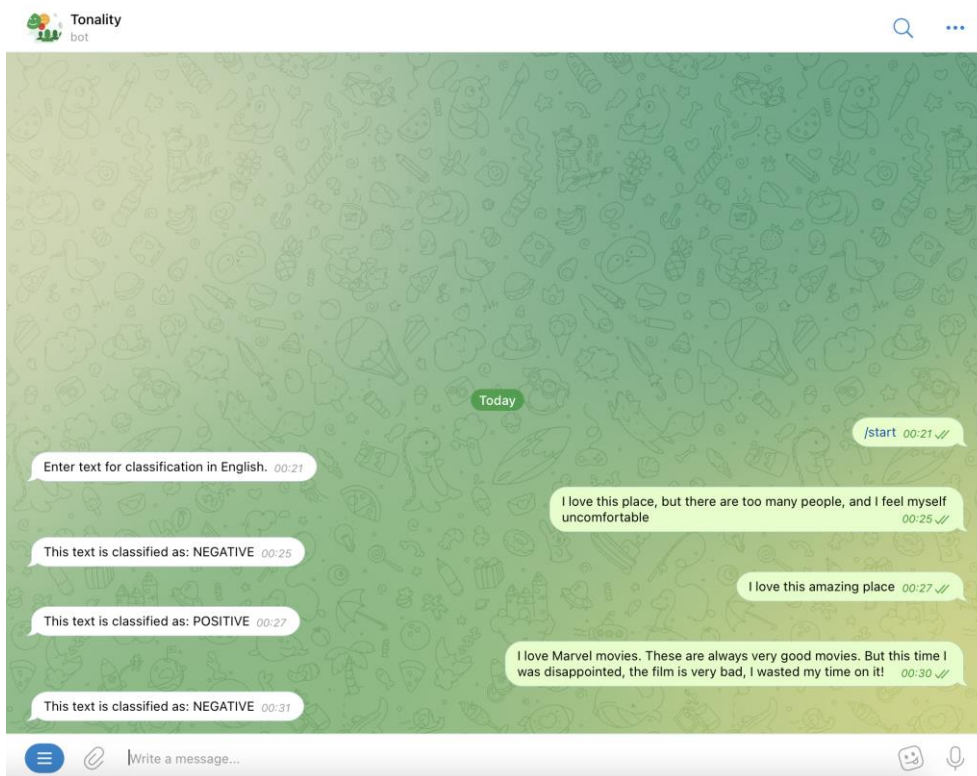


Рисунок - 3.9 Результати класифікації



Рисунок - 3.10 Результати класифікації

Можемо бачити, що бот чудово справляється з таким видом текстів також.

Висновки до розділу 3

Використання команди /start для активації бота є стандартною процедурою в багатьох телеграм ботах, що робить його зручним і легким у використанні для кінцевого користувача.

Процес взаємодії з ботом починається з відправлення тексту, після чого бот проводить його аналіз і відправляє результати класифікації у вигляді повідомлення. Ця властивість бота демонструє його здатність до швидкого та точного аналізу введених даних. Особливо важливо, що бот ефективно справляється з текстами різної довжини та складності, включаючи повідомлення, що складаються з декількох речень. Це свідчить про високу адаптивність і гнучкість алгоритмів, що лежать в основі бота.

Таким чином розроблений бот забезпечує користувачам швидкий, зручний та точний спосіб аналізу емоційного забарвлення текстів у Telegram. Це робить його цінним інструментом як для особистих, так і для бізнес-застосувань, де аналіз настроїв може відігравати ключову роль у прийнятті рішень, маркетингових стратегіях та обслуговуванні клієнтів.

ВИСНОВКИ

Аналіз настроїв у сучасному світі набуває особливої важливості, оскільки він знаходить застосування у різноманітних сферах, включаючи прогнозування фондового ринку, обслуговування клієнтів та маркетинг. Завдяки прогресу в області обробки природної мови (NLP) і машинного навчання, відкриваються нові можливості для розуміння та аналізу емоційного змісту текстів. Різні методи, такі як згорткові нейронні мережі, рекурентні нейронні мережі та двонаправлені трансформатори Бєрта, використовуються для досягнення цієї мети, кожен з яких має свої особливості, рівні точності та швидкості обчислень.

Незважаючи на значні досягнення в сфері аналізу настроїв, існує брак інструментів, доступних для широкого кола користувачів, особливо у соціальних мережах як Telegram. Ця дисертація заповнює цю прогалину, розробляючи та впроваджуючи модель, засновану на передових методах машинного навчання для аналізу настроїв текстів у Telegram. Реалізація включає в себе використання Python 3, методу Бєрта та бібліотеки TelegramBotApi, що забезпечує гнучкість і масштабованість рішення.

Проектування та розробка Telegram-бота з акцентом на аналіз настроїв включає ретельне планування його структури та архітектури. Це забезпечує ефективну обробку різних типів повідомлень, від простих до складних, забезпечуючи високу точність результатів незалежно від кількості речень у повідомленні. Важливим аспектом є також забезпечення здатності бота оновлювати свої методи аналізу, що робить його адаптивним до змін у мовних тенденціях і стилях спілкування.

Загальні висновки дослідження підтверджують, що інтеграція передових методів машинного навчання для аналізу настроїв у практичні застосування, як-от Telegram-бот, є ефективною та інноваційною. Це не лише розширює можливості для бізнесу та особистого використання, але й стимулює подальший розвиток технологій

обробки природної мови. Результати цієї роботи можуть служити основою для майбутніх досліджень та розробок у цій динамічно зростаючій області.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. SentimentAnalysis:ADefinitiveGuide [Електронний ресурс] - Режим доступу до ресурсу: <https://monkeylearn.com/sentiment-analysis/>
2. SentimentAnalysis:Types,Tools,andUseCases [Електронний ресурс] - Режим доступу до ресурсу: <https://www.altexsoft.com/blog/business/sentiment-analysis-types-tools-and-use-cases/>
3. SentimentAnalysisinBanking—4CurrentUse-Cases [Електронний ресурс]- Режим доступу до ресурсу: <https://emerj.com/ai-sector-overviews/sentiment-analysis-banking/>
4. Аналіз тональності тексту [Електронний ресурс]- Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Аналіз_тональності_тексту
5. Вступ до машинного навчання [Електронний ресурс]- Режим доступу до ресурсу: <http://specials.kunsht.com.ua/machinelearning2>
6. BERT (модель мови) [Електронний ресурс]- Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/BERT_\(модель_мови\)](https://uk.wikipedia.org/wiki/BERT_(модель_мови))
7. BERT Explained: State of the art language model for NLP [Електронний ресурс]- Режим доступу до ресурсу: <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>
8. Word2vec [Електронний ресурс]- Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/Word2vec>
9. Применение сверточных нейронных сетей для задач NLP [Електронний ресурс]- Режим доступу до ресурсу: <https://habr.com/ru/company/ods/blog/353060/>
10. Рекуррентна нейронна мережа [Електронний ресурс]- Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Рекуррентна_нейронна_мережа
11. Text Classification with RNN [Електронний ресурс]- Режим доступу до ресурсу: <https://towardsai.net/p/deep-learning/text-classification-with-rnn>

12. Модель RoBERTa [Электронный ресурс]- Режим доступа до ресурсу:
<https://huggingface.co/siebert/sentiment-roberta-large-english>
13. To Pretrain or Not to Pretrain: Examining the Benefits of Pretraining on Resource Rich Tasks [Электронный ресурс]- Режим доступа до ресурсу:
<https://www.aclweb.org/anthology/2020.acl-main.200.pdf>
14. RoBERTa: A Robustly Optimized BERT Pretraining Approach [Электронный ресурс]- Режим доступа до ресурсу: <https://arxiv.org/pdf/1907.11692.pdf>

ДОДАТОК

```
from telegram.ext import Updater, CommandHandler, MessageHandler,
Filters
from transformers import pipeline

# Initialize sentiment analysis model
sentiment_model = pipeline("sentiment-analysis", model="distilbert-
base-uncased-finetuned-sst-2-english")

# Function to analyze sentiment
def analyze_sentiment(update, context):
    user_text = update.message.text
    result = sentiment_model(user_text)[0]
    sentiment = "positive" if result['label'] == 'POSITIVE' else
"negative"
    score = result['score']
    response = f"Your message seems to have a {sentiment} sentiment
with a confidence score of {score:.2f}."
    update.message.reply_text(response)

# Function to start the bot and greet the user
def start(update, context):
    update.message.reply_text("Hello! I'm your sentiment analysis bot.
Send me a message, and I'll analyze its sentiment.")

def main():
    # Telegram bot token (replace with your own token)
    token = '5356003407:AAF5X3zOMj_k6cFwCHsD4vuWufebZYaqmB0'

    # Create the Updater and pass it your bot's token
    updater = Updater(token, use_context=True)

    # Get the dispatcher to register handlers
```

```
dispatcher = updater.dispatcher

# Command handler for starting the bot
start_handler = CommandHandler('start', start)
dispatcher.add_handler(start_handler)

# Message handler for processing text messages
text_handler = MessageHandler(Filters.text & ~Filters.command,
analyze_sentiment)
dispatcher.add_handler(text_handler)

# Start the Bot
updater.start_polling()
updater.idle()

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