Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет _____Комп'ютерних наук_____
(повна назва)
Кафедра _____Програмної інженерії_____
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

_____другий (магістерський)_____
(рівень вищої освіти)

Дослідження методів аналізу тональностей тексту
(тема)

Виконав: студент 2 курсу, групи ІПЗм-17-1_
спеціальності 121- Інженерія програмного забезпечення
(код і повна назва спеціальності)
Освітньо-професійної програми
Інженерія програмного забезпечення
_____Бабаскіна І.О_____
(прізвище, ініціали)
Керівник _____доц. Валєнда Н.А._____
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри, проф. _____ З.В.Дудар

2019 р.

Харківський національний університет радіоелектроніки

Факультет комп'ютерних наук
Кафедра програмної інженерії
Рівень вищої освіти другий (магістерський)

Спеціальність 121– Інженерія програмного забезпечення
(код і повна назва)
Освітньо-професійна програма Програмне забезпечення систем
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____»_____ 20 ___ р.

ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУ

Студентові Бабаскінй Ірині Олегівні
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів аналізу тональностей тексту

затверджена наказом по університету від «09» листопада 2018 р № 1592 Ст

2. Термін подання студентом роботи до екзаменаційної комісії «17» січня 2019 р.

3. Вихідні дані до роботи *порівняльний аналіз методів, претреновані моделі нейронних мереж, пояснювальна записка. Використовувати ОС Windows, середовище об'єктно-орієнтованого проектування*

4. Перелік питань, що потрібно опрацювати в роботі *позначка роботи, аналіз проблемної галузі і постановка задачі, опис проблем продуктивності, використовувані методи та алгоритми, опис розробленої програмної системи, аналіз можливих застосувань*

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) *Мета завдання, обґрунтування доцільності розроблення, постановка задачі, досліджувані методи, базові моделі, графічний інтерфей системи, результати тестування програмної системи, демонстраційні матеріали*

6 Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім’я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	доц. Валенда Н.А.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі		
2.	Огляд існуючих методів		
3.	Методи підвищення ефективності ве додатків		
4.	Підготовка пояснювальної записки		
5.	Спецчастина		
6.	Підготовка презентації та доповіді		
7.	Попередній захист		
8.	Нормоконтроль, рецензування		
9.	Занесення диплома в електронний архів		
10	Допуск до захисту у зав. кафедри		
* заповнюється вручну після виконання чергового пун			

Дата видачі завдання _____ _____2019 р.

Студент _____
(підпис)

Керівник роботи _____ __доц. Валенда Н.А._____

РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи: 62 с., 11 рис., 1 таблиця, 3 додатки, 21 джерел.

АНАЛІЗ ТОНАЛЬНОСТІ, НЕЙРОНА МЕРЕЖА, МАШИННЕ НАВЧАННЯ, PYTHON, ТОНАЛЬНІСТЬ ТЕКСТУ, РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, ТРАНСФЕРНЕ НАВЧАННЯ

Об’єктом дослідження є тональність (емоційне забарвлення) текстів природної мови. Предмет дослідження – існуючі методи аналізу тональності тексту, що базуються на використанні нейронних мереж; також використовується підхід трансферного навчання, і в результаті будуються попередньо навчені моделі для розв’язання задачі визначення тональності тексту.

Методи розробки базуються на наступних мовах та технологіях: Python, Keras, TensorFlow

У результаті роботи побудовано та натреновано моделі, що можуть використовуватися для подальшого аналізу тональностей текстів, та сконструйовано веб-додаток для визначення тональності довільного тексту.

SENTIMENT ANALYSIS, NEURAL NETWORK, MACHINE LEARNING, PYTHON, TONNALITY OF TEXT, RECURENT NEURAL NETWORKS, CONVOLUTIONAL NEURAL NETWORKS, TRANSFER LEARNING

The object of the research is the a sentiment analysis of the natural language text. Subject of research - existing methods of sentiment analysis, based on the use of neural

networks; Also, the transfer training approach is used, and as a result, pre-trained models are developed for solving the task of determining the tonality of the text.

Development methods are based on the following languages and technologies: Python, Keras, TensorFlow

As a result of the work, constructed and trained models that can be used for further analysis of the tonality of texts, and a web application designed to determine the tonality of arbitrary text.

ЗМІСТ

ВСТУП

Про обробку природної мови сьогодні багато говорять – причому, не тільки в наукових колах, де ця концепція справедливо вважається основним для подальшого розвитку штучного інтелекту, а й представників IT-індустрії.

Серед найбільш цікавих і популярних методів цього широкого наукового напрямку осібно стоїть один, що носить назву sentiment analysis, аналіз тональності текстів. Загальне визначення свідчить, що аналіз тональності текстів – це клас методів контент-аналізу, призначений для автоматичного виявлення в тексті емоційно забарвленої лексики, а також думок (емоційних оцінок) автора з приводу об'єктів, в яких йде мова в тексті. З визначення можна зробити кілька висновків про те, де концепція аналізу тональності тексту могла б знайти застосування і прояснити деякі її деталі.

По-перше, аналіз тональності текстів здатний допомогти розібратися в законах, за якими живе природна мова і навчити комп'ютер сприймати його на рівні, наближеному до людського. До недавнього часу машина розуміла тексти на абстрактному рівні - в основному, через лексеми (слова), які для неї мали формою (набір букв) і змістом (значення). Дана концепція пропонує ввести ще одну функцію – так звану лексичну тональність тексту (в найпростішому випадку вона буде визначатися як сума лексичних тональностей кожної окремої лексеми, з яких складається текст).

По-друге, аналіз тональності здатний значно покращити якість перекладів. Відомо, що еталоном машинного перекладу служить результат перекладу тексту людиною - професійним перекладачем. За більше ніж п’ятдесят років розробок в цій області дослідники переконалися в тому, що навчити машину «думати, як перекладач» можна лише взявши до уваги всі ті міркування, якими користується професіонал, переводячи той чи інший текст. При перекладі не обійтися без первинного аналізу тексту та окремих слів - в тому числі, аналізу тональності як такої.

По-третє, метою аналізу тональності тексту може бути якесь думка автора або сам автор. Це - найбільш цікава сфера застосування, оскільки тут бачиться не тільки спосіб делегування машині деяких повноважень вченого (наприклад, філолога, який досліджує твір того чи іншого автора), але і знову спроба наблизити образ мислення комп'ютера до людського. З цієї точки зору аналіз тональності, можливо, є одним з найбільш важливих і перспективних кроків до розвитку штучного інтелекту.

Отже, метою роботи є дослідження існуючих методів аналізу тональності тексту та їх різновидів і виявлення найбільш ефективних методів. Особлива увага в дослідженні буде приділятися використанню нейронних мереж та машинного навчання для розв’язування задачі аналізу тональності заданого тексту.

Об’єктом дослідження є тональність (емоційне забарвлення) текстів природної мови. Предмет дослідження – існуючі методи аналізу тональності тексту, що базуються на використанні нейронних мереж; також використовується підхід трансферного навчання, і в результаті будуються попередньо навчені моделі для розв’язання задачі визначення тональності тексту.

Результати проведеного дослідження може бути використано для розв’язання прикладних задач, як теоретичні результати (порівняння різних методів), так і практичні – попередньо навчені моделі, що можуть бути використані безпосередньо для визначення тональності тексту без необхідності будувати і тренувати нову нейронну мережу.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз тональності текстів

Аналіз тональності тексту (сентимент-аналіз, англ. Sentiment analysis, англ. Opinion mining) – це клас методів аналізу контенту в комп'ютерній лінгвістиці, що призначений для автоматизованого виявлення в текстах емоційно забарвленої лексики і емоційної оцінки автора тексту по відношенню до об'єктів, мова про які йде в тексті[1].

Тональність – це ставлення автора висловлювання до того, про що йде мова в тексті (об'єкту реального світу, події, процесу або їх властивостей), виражене в тексті. Емоційна складова, виражена на рівні лексеми або комунікативного фрагмента, називається лексичної тональністю (або лексичним сентиментом). Тональність всього тексту в цілому можна визначити як функцію (в найпростішому випадку суму) лексичних тональностей складових його одиниць (речень) і правил їх поєднання.

Аналіз тональності текстів – це поле для обробки природних мов (Natural language processing, NLP), яке будує системи, які намагаються ідентифікувати і витягти думки в тексті[2]. Зазвичай, крім ідентифікації думки, ці системи витягують атрибути виразу, наприклад:

– полярність: оратор висловлює позитивну чи негативну думку;

– тема: про те, про що говорять;

– власник думки: особа або організація, яка висловлює свою думку.

В даний час аналіз настроїв є цікавим напрямом, що розвивається, оскільки має багато практичних застосувань. Оскільки публічно та приватно доступна інформація по Інтернету постійно зростає, велика кількість текстів, що висловлюють свої думки, доступна на сайтах рецензування, форумах, блогах і соціальних медіа.

За допомогою систем аналізу настроїв ця неструктурована інформація може автоматично трансформуватися у структуровані дані громадської думки про

продукти, послуги, бренди, політику або будь-яку тему, щодо якої люди можуть висловлювати свої думки. Ці дані можуть бути дуже корисними для комерційних додатків, таких як маркетинговий аналіз, зв'язки з громадськістю, огляди продуктів, підрахунок чистого промоутера, відгуки про продукцію та обслуговування клієнтів.

Текстову інформацію можна розділити на два основні типи: факти та думки[1]. Факти є об'єктивними виразами про щось. Думки - це, як правило, суб'єктивні вирази, що описують почуття, оцінки та почуття людей до теми чи теми.

Аналіз сентиментів, як і багато інших задач NLP, може бути змодельований як проблема класифікації, де необхідно вирішити двіі підпроблеми:

– класифікація речення як суб'єктивна або об'єктивна, відома як класифікація суб'єктності;

– класифікація речення як вираження позитивного, негативного або нейтрального думки, відома як класифікація полярності.

На думку, об'єкт, про який говорить текст, може бути об'єктом, його складовими, його аспектами, його атрибутами або його особливостями. Це також може бути продукт, послуга, особа, організація, подія або тема. Наприклад: "Термін служби акумулятора цієї камери надто короткий" – висловлюється негативна думка про особливість (час автономної роботи) суб'єкта (камери).

Існує два види думок: прямий і порівняльний. Прямі висновки дають висновок про суб'єкт безпосередньо, наприклад: "Якість зображення камери А погана.". Ця пряма думка висловлює негативну думку про камеру А.

У порівняльних висновках думка виражається шляхом порівняння суб'єкта, про який йдеться мова в тексті, з іншим, часто таким, що належить до такого ж классу, наприклад: "Якість зображення камери А краще, ніж у камери В.". Як правило, порівняльні думки виражають подібності або відмінності між двома або більше суб'єктами, використовуючи порівняльну або чудову форму прикметника або прислівника. У попередньому прикладі існує позитивна думка про камеру А і, навпаки, негативна думка про камеру Б.

Явна думка по темі - це думка, явно виражена в суб'єктивному реченні. Наступне речення виражає явну позитивну думку: "Якість голосу цього телефону дивовижна.".

Імпліцитна думка з цього питання є думкою, що передбачається в об'єктивному реченні. Наступне речення виражає неявну негативну думку: "Наушник зламався через два дні". У неявних думках ми могли б включати метафори, які можуть бути найскладнішим типом думок для аналізу, оскільки вони містять багато семантичної інформації[3].

1.2 Типи аналізу тональностей

В сучасних системах автоматичного визначення емоційної оцінки тексту найчастіше використовується одномірний емотивний простір: позитив чи негатив (добре або погано). Однак відомі успішні випадки використання і багатовимірних просторів.

Основним завданням в аналізі тональності є класифікація полярності текста, тобто визначення, чи є виражена в тексті думка позитивною, негативною або нейтральною. Більш розгорнута класифікація тональності виражається, наприклад, такими емоційними станами, як «злий», «сумний» і «щасливий».

Полярність документа можна визначати за бінарною шкалою. У цьому випадку для визначення полярності документа використовується два класи оцінок: позитивна чи негативна.

Одним з недоліків цього підходу є те, що емоційну складову документа не завжди можна однозначно визначити, тобто документ може містити ознаки позитивної оцінки, так і негативної ознаки. Ранні роботи в цій області включають в себе праці Терні і Панга, які застосовують різні методи розпізнавання полярності оглядів товару і відгуків про фільмах відповідно. Це приклад роботи на рівні документа.

Можна класифікувати полярність документа по багатосмуговій шкалою, що було зроблено Пангом і Снайдером (серед інших)[2]. Ними було розширене основне завдання класифікації кіновідгуків від оцінки «позитивний або негативний» в бік прогнозування рейтингу по 3-х або 4-бальною шкалою. У той же час Снайдер провів поглиблений аналіз оглядів ресторанів, пророкуючи рейтинги їх різних властивостей, таких як їжа і атмосфера (за 5-бальною шкалою).

Іншим методом визначення тональності є використання систем шкалювання, за допомогою чого словами, зазвичай пов'язаних з негативними, нейтральними або позитивними тональностями, ставляться відповідно числа за шкалою від -10 до 10 (від негативного до самого позитивного). Спочатку фрагмент неструктурованого тексту досліджується з допомогою інструментів та алгоритмів обробки природної мови, а потім виділені з цього тексту об'єкти та терміни аналізуються з метою розуміння значення цих слів.

Інший дослідницький напрямок – це ідентифікація суб'єктивності/об'єктивності. Це завдання зазвичай визначається як віднесення даного тексту в один з двох класі суб'єктивний й або об'єктивний. Ця проблема іноді може бути більш складною, ніж класифікація полярності: суб'єктивність слів і фраз може залежати від контексту, а об'єктивний документ може містити в собі суб'єктивні пропозиції (наприклад, новинна стаття, цитує думки людей). Більш того, як згадував Су[4], результати більшою мірою залежать від визначення суб'єктивності, вживаючийся в рамках анотації текстів. Як би те ні було, Панг показав, що видалення об'єктивних пропозицій з документа перед класифікацією полярності допомогло підвищити точність результатів.

Модель більш докладного аналізу називається аналізом на основі функції/аспекту. Ця модель посилається на ухвалу думок або настроїв, виражених різними функціями або аспектами сутностей, наприклад, у стільникового телефону, цифрової камери або банку. Властивість/аспект – це атрибут або компонент сутності, досліджуваної на тональність, наприклад, екран мобільного телефону або ж якість зйомки камери. Ця проблема вимагає вирішення ряду завдань, наприклад, ідентифікація актуальних сутностей, витяг їх функцій аспектів

та визначення, є думка, висловлена по кожній функції/аспекту, позитивним, негативним або нейтральним. Більш докладні дискусії на цей рахунок можуть бути знайдені в довіднику з NLP, у главі «Аналіз тональності та суб'єктивності»[5].

Існує багато видів аналізу настроїв, а інструменти аналізу тональностей варіюються від систем, які зосереджуються на полярності (позитивні, негативні, нейтральні) до систем, які виявляють почуття і емоції (сердиті, щасливі, сумні тощо) або визначають наміри (наприклад, зацікавлені v. не зацікавлений). У наступному розділі ми розглянемо найважливіші з них.

Інколи ви можете бути більш точними щодо рівня полярності думки, тому замість того, щоб просто говорити про позитивні, нейтральні або негативні думки, можна розглянути такі категорії:

- дуже позитивно;
- позитивний;
- нейтральний;
- негативний;
- дуже негативний.

Це зазвичай називають дрібнозернистим аналізом. Це може бути, наприклад, нанесено на 5-зірковий рейтинг в огляді, наприклад: Дуже позитивний = 5 зірок і Дуже негативний = 1 зірка.

Деякі системи також надають різні відтінки полярності, визначаючи, чи позитивні або негативні настрої асоціюються з певним почуттям, таким як гнів, смуток або турботи (тобто негативні почуття) або щастя, любов або ентузіазм (тобто позитивні почуття).

Виявлення емоцій спрямоване на виявлення таких емоцій, як щастя, розчарування, гнів, смуток і тому подібне. Багато систем виявлення емоцій вдаються до лексиконів (тобто списків слів і емоцій, які вони передають) або складних алгоритмів машинного навчання.

Одним з недоліків вживання лексиконів є те, що спосіб, у який люди висловлюють свої емоції, сильно змінюється, так само як і лексичні предмети, які вони використовують. Деякі слова, які, як правило, виражають гнів, як лайно або

вбивають (наприклад, у вашому продукті це лайно, або ваша клієнтська підтримка вбиває мене), можуть також виражати щастя (наприклад, у таких текстах, як Це лайно, або Ви його вбиваєте).

Як правило, при аналізі настроїв у суб'єктах, наприклад, на продуктах, вас може зацікавити не тільки те, що люди говорять з позитивною, нейтральною або негативною полярністю про продукт, а й які конкретні аспекти або особливості продукту, про який люди говорять . Ось про що йдеться в аспект-аналізі. У нашому попередньому прикладі: "Термін служби акумулятора цієї камери надто короткий".

Пропозиція висловлює негативну думку про камеру, а точніше, про час автономної роботи, що є особливою особливістю камери.

Аналіз наміру в основному виявляє те, що люди хочуть робити з текстом, а не те, що люди говорять з цим текстом. Подивіться на наступні приклади.

Приклад перший: “Ваша підтримка клієнтів є катастрофою. Я тримаюся 20 хвилин”.

Приклад другий: “Я хотів би знати, як замінити картридж”.

Приклад третій: "Чи можете ви допомогти мені заповнити цю форму?"

Людина не має проблем з виявленням скарги в першому тексті, питанням у другому тексті, а запит в третьому тексті. Тим не менш, машини можуть мати деякі проблеми, щоб визначити їх. Іноді передбачувана дія може бути виведена з тексту, але іноді виведення цього вимагає певного контекстуального знання, для якого шуточному інтелекту потрібна додаткова інформація.

Багатомовний аналіз настроїв може бути складним завданням. Як правило, необхідна велика кількість попередньої обробки, і попередня обробка використовує ряд ресурсів. Більшість цих ресурсів доступні в Інтернеті (наприклад, лексикони сентиментів), але потрібно створити багато інших (наприклад, перекладені корпуси або алгоритми виявлення шуму). Використання доступних ресурсів вимагає багато досвіду кодування і може зайняти багато часу для реалізації.

1.3 Існуючі методи аналізу тональностей

Існує багато методів і алгоритмів для реалізації систем аналізу настроїв, які можна класифікувати як:

– системи на основі правил, які виконують аналіз настроїв на основі набору ручних правил;

– автоматичні системи, які спираються на методи машинного навчання, щоб дізнатися з даних;

– гібридні системи, які поєднують обидві правила і автоматичні підходи.

Як правило, підходи, що базуються на правилах, визначають набір правил у певній скриптовій мові, яка ідентифікує суб'єктність, полярність або предмет думки.

Правила можуть використовувати різні входи, такі як:

– класичні методи НЛП, такі як витіснення, маркування, частина мічення мовлення і розбору;

– інші ресурси, такі як лексикони (тобто списки слів і виразів).

Основним прикладом реалізації на основі правил буде наступне:

Визначте два списки поляризованих слів (наприклад, негативні слова, такі як погані, найгірші, потворні тощо, і позитивні слова, такі як хороші, найкращі, красиві тощо).

Дано текст.

Підрахуйте кількість позитивних слів, які з'являються в тексті.

Підрахуйте кількість негативних слів, які з'являються в тексті.

Якщо кількість позитивних виступів слів більше, ніж кількість негативних слів, то повернення позитивних настроїв, навпаки, повернення негативних настроїв. В іншому випадку поверніть нейтраль.

Ця система дуже наївна, оскільки не враховує, як слова поєднуються в послідовності. Можна зробити більш просунуту обробку, але ці системи швидко стають дуже складними. Їх дуже важко підтримувати, оскільки можуть

знадобитися нові правила, щоб додати підтримку для нових виразів і словника. Крім того, додавання нових правил може мати небажані наслідки в результаті взаємодії з попередніми правилами. Як результат, ці системи вимагають важливих інвестицій в ручне налаштування та підтримку правил.

Автоматичні методи, всупереч системам на основі правил, не покладаються на правила, створені вручну, а на техніку машинного навчання. Завдання аналізу настроїв зазвичай моделюється як проблема класифікації, коли класифікатор подається текстом і повертає відповідну категорію, наприклад, позитивний, негативний або нейтральний (у випадку, якщо проводиться аналіз полярності).

Перший крок у класифікаторі тексту машинного навчання полягає в перетворенні тексту в числове представлення, зазвичай вектор. Зазвичай кожен компонент вектора являє частоту слова або виразу в попередньо визначеному словнику (наприклад, лексикон поляризованих слів). Цей процес відомий як екстракція ознак або векторизація тексту, а класичний підхід був мішкою слів або мішкою з їх частотою.

Зовсім недавно були застосовані нові методи вилучення ознак на основі вбудовування слів (також відомих як вектори слів). Таке уявлення дає можливість для слів з подібним значенням мати подібне уявлення, що може поліпшити продуктивність класифікаторів.

Крок класифікації, як правило, включає статистичну модель, таку як Naive Bayes, логістична регресія, машини підтримки векторних чи нейронних мереж.

1.4 Метрики аналізу тональностей текстів

Існує багато способів, за допомогою яких можна отримати показники ефективності для оцінки класифікатора і зрозуміти, наскільки точна модель аналізу настроїв. Один з найбільш часто використовуваних називається перехресною перевіркою.

Що робить перехресна перевірка - це поділ тренувальних даних на певну кількість складних даних (з 75% навчальних даних) і таку ж кількість тестових даних (з 25% навчальних даних), використання тренувальних даних для навчання Класифікатора, і перевірити його на тестування складок, щоб отримати показники продуктивності. Процес повторюється кілька разів і обчислюється середня для кожної метрики.

Якщо тестовий набір завжди один і той же, ви перенасилити тестовий набір, що означає, що ви можете скоригувати свій аналіз для даного набору даних так, що ви не зможете проаналізувати інший набір. Перехресна перевірка допомагає запобігти цьому[3]. Чим більше даних, тим більше різних наборів ви зможете використовувати.

Точність, продуктивність та час відклику є стандартними показниками, які використовуються для оцінки продуктивності класифікатора.

Точність вимірює, скільки текстів було передбачено правильно як приналежність до даної категорії з усіх текстів, які були передбачені (правильно і неправильно) як належать до категорії.

Нагадаємо про міри, скільки текстів було передбачено правильно як приналежність до даної категорії з усіх текстів, які повинні були передбачати, що належать до категорії. Ми також знаємо, що чим більше даних ми будемо годувати нашими класифікаторами, тим краще буде згадувати.

Найчастіше для вимірювання продуктивності використовують точність і час відклику, оскільки тільки точність не говорить про те, наскільки хорошим чи поганим є класифікатор.

Для складних завдань, таких як аналіз настроїв, точність і час відклику, напевно, будуть низькими. Після подання класифікатору більше даних продуктивність буде покращена. Проте, як ми побачимо нижче, оскільки анотовані дані навряд чи будуть точними, є ймовірність того, що рівні точності не стануть надто високими. Однак, якщо ви подаєте класифікатору послідовно розмічені даних, результати будуть настільки ж хорошими, як результати можуть бути для будь-якої іншої проблеми класифікації.

Коли мова йде про угоду між анотаторами (тобто домовленість людей щодо даного завдання анотації), однією з найбільш часто використовуваних показників є криппендорфська альфа. На думку Сайфа та ін., найкраща угода між анотаторами для аналізу настроїв у Twitter досягає значення 0.655 для Alpha у Krippendorff. Це означає, що є багато угод (оскільки альфа більше нуля), але ми вважаємо, що це ще далеко від великого (наприклад: близько 0,8, що є мінімальним порогом надійності, яке використовують соціологи для того, щоб сказати, що дані є надійними).

1.5 Використання аналізу тональностей текстів

1.5.1 Аналіз настроїв у моніторингу соціальних медіа

У фатальний вечір 9 квітня 2017 року United Airlines примусово вивезла пасажира з перенавантаженого рейсу. Інцидент з кошмаром був знятий іншими пасажирами на їхніх смартфонах і негайно розміщений[6]. Одне з таких відео, розміщене на Facebook, було більше 87 тисяч разів і було переглянуто 6,8 мільйона разів у 6 вечора в понеділок, лише через 24 години.

Фіаско жахливо звеличувалося завдяки репрезентативній реакції компанії. У понеділок після обіду вони надіслали заяву від генерального директора про вибачення за те, що їм доведеться повторно розмістити клієнтів.

Це саме той тип PR-катастрофи, без якого ми всі хотіли б щасливо працювати. Це також чудовий приклад того, чому ми піклуємося не тільки про те, що люди говорять про наш бренд, але й про те, як вони це говорять. Більше згадувань не дорівнює позитивним згадкам.

У сьогоднішній добі і сторіччі, бренди всіх форм і розмірів мають значущі взаємодії з клієнтами, провідниками і навіть конкуренцією в соціальних мережах, таких як Facebook, Twitter і Instagram. Більшість маркетингових департаментів вже налаштовані на онлайн-згадки щодо обсягу - вони оцінюють більше балачки, оскільки більше знань про бренд. Сьогодні, однак, ми можемо зробити ще глибше.

Використовуючи аналіз настроїв у соціальних мережах, ми можемо отримати неймовірну інформацію про якість розмови, що відбувається навколо бренду.

Як можна використовувати аналіз настроїв:

– проаналізуйте твіти та / або повідомлення на Facebook протягом періоду часу, щоб побачити почуття певної аудиторії;

– запустіть аналіз настроїв у всіх згаданих соціальних мережах до вашого бренду та автоматично класифікуйте їх за терміном;

– автоматично направляйте повідомлення про соціальні медіа для членів команди, які найкраще відповідатимуть;

– автоматизуйте будь-який або всі ці процеси;

– використовуйте аналітику, щоб отримати глибоке розуміння того, що відбувається на ваших каналах соціальних медіа.

Переваги використання аналізу тональностей тексту для моніторингу соціальних мережах:

– визначення пріоритетністі дій. Що більш терміново: димлячий клієнт або тонке "спасибі!" Очевидно, споживач. Аналіз настроїв дозволяє легко фільтрувати непрочитані згадки за допомогою позитивності та негативу, показуючи вам, які палаючі вогні покласти на список "негайно погасити" і які повільні тлеючі можуть трохи почекати;

– відстежування тенденції з плином часу;

– визначення найбільш прийнятного моменту для запуску нового продукту чи відстеження зменшення обсягу продаж;

– відстеження соціальних мереж конкурентів, що може допомогти, наприклад, не повторювати помилок конкурентів (якщо якийсь новий продукт покупцям не сподобався або рекламна компанія, наприклад, виявилася невдалою, цей досвід можна використати):

– використовуйте аналітику, щоб отримати глибоке розуміння того, що відбувається на ваших каналах соціальних медіа.

1.5.2 Аналіз настроїв у моніторингу брендів

Не тільки бренди мають багату інформацію, доступну в соціальних мережах, але вони також можуть виглядати більш широко в Інтернеті, щоб побачити, як люди говорять про них в Інтернеті. Замість того, щоб зосередитися на певних соціальних медіа-платформах, таких як Facebook і Twitter, ми можемо націлювати згадки в таких місцях, як новини, блоги та форуми - знову, не тільки обсяг згадувань, а й якість цих згадок.

У прикладі United Airlines, наприклад, почалася розгорання ситуації на платформах соціальних медіа декількох пасажирів. Протягом декількох годин він був підхоплений новинними сайтами і поширився як лісова пожежа по США. Потім новини поширилися в Китай і В'єтнам, оскільки пасажир, як повідомлялося, був американцем китайсько-в'єтнамського походження, і люди звинувачували винних у расовому профілюванні. У Китаї інцидент став трендовою темою номер один на сайті Weibo, мікроблогів з майже 500 мільйонами користувачів[7].

І знову ж таки, все це відбувається протягом кількох годин і днів, коли стався інцидент.

Як можна використовувати аналіз настроїв:

– проаналізуйте новини, публікації в блогах, обговорення на форумі та інші тексти в Інтернеті протягом певного періоду часу, щоб побачити почуття певної аудиторії;

– автоматично класифікуйте терміновість всіх онлайнових згадок вашого бренда через аналіз настроїв;

– автоматично сповіщати призначених членів групи про онлайн-згадки, які стосуються їхньої сфери діяльності;

– автоматизуйте будь-який або всі ці процеси;

– краще зрозумійте присутність бренду в Інтернеті, отримуючи всілякі цікаві ідеї та аналітику для подальшого розвинення бренду, беручі за основу справжні клієнтські відгуки.

Аналіз спостережень корисний для моніторингу бренду, оскільки він допомагає виконувати наступне:

– розуміння, як з часом розвивається репутація марки;

– дослідження конкурентів і розуміння, як з часом змінюється їхня репутація;

– визначення потенційні кризи в сфері зв'язків з громадськістю і знання, що необхідно вжити негайних заходів;

– фокус на певному момент часу. Знову ж таки, можливо, ви хочете подивитися лише згадки про пресу в день подачі заявки на IPO, або запуск нового продукту. Аналіз настроїв дозволяє це зробити.

Приклад: Expedia Canada. Біля Різдва, Expedia Canada провела класичну маркетингову кампанію "Зимова втеча". Все було добре, за винятком вибору скрипки як фонової музики. Зрозуміло, що люди потрапили до соціальних медіа, блогів і форумів. Expedia помітив і видалив оголошення. Потім вони створили серію відеозаписів: один показав, що оригінальний актор розбив скрипку, а в іншому – запросив справжнього послідовника, який скаржився в Twitter, щоб він прийшов і зірвав скрипку.

Хоча їх оригінальний продукт був далеко не бездоганним, врешті решт вони змогли викупити себе, включивши реальні відгуки клієнтів до продовжених ітерацій.

Використовуючи аналіз настроїв (і машинне навчання), ви можете автоматично контролювати всі розмови та відгуки навколо свого бренду і виявляти цей тип потенційно-вибухового сценарію, поки у вас ще є час для його розрядки[8].

1.5.3 Аналіз настроїв у відгуках клієнтів

Соціальні медіа та моніторинг бренду пропонують нам негайну, нефільтровану, безцінну інформацію про настрої клієнтів. У паралельному ключі запускаються дві інші команди розуміння - опитування та взаємодія з клієнтською

підтримкою. Команди часто дивляться на свій чистий показник промоутера (NPS), але ми також можемо застосувати цей аналіз до будь-якого типу обстеження або каналу зв'язку, що дає можливість отримати текстові відгуки клієнтів.

Опитування NPS задають кілька простих запитань, а саме: чи рекомендували б ви цю компанію, продукт і / або послугу своєму другові або члену сім'ї? і чому? - і використовувати це для ідентифікації клієнтів як промоутерів, пасивів або недоброзичливців.

Мета полягає в тому, щоб визначити загальний досвід клієнтів, і знайти способи підняти всіх клієнтів на рівень «промоутер», де вони теоретично купуватимуть більше, залишатимуться довше і направлятимуть інших клієнтів.

Дані чисельних опитувань легко агрегуються та оцінюються, але ми хочемо, щоб таку ж легкість відповідали і на питання «чому». Регулярний показник NPS просто дає вам номер, без додаткового контексту того, про що йде мова і чому оцінка приземлилася там. Аналіз сприйняття робить цей крок подальшим.

Як можна використовувати аналіз настроїв:

– аналіз агрегованих NPS або інших відповідей на дослідження;

– проаналізуйте агреговані взаємодії з клієнтською підтримкою;

– відстежуйте настрої клієнтів щодо конкретних аспектів бізнесу з плином часу. Це додає глибину, щоб пояснити, чому загальна оцінка NPS може змінитися, або якщо окремі аспекти змінилися незалежно;

– орієнтуйте людей на поліпшення їхнього обслуговування. Автоматизуючи аналіз настроїв на вхідні опитування, ви можете бути попереджені клієнтам, які відчувають негативний вплив на ваш продукт або послугу, і можуть конкретно займатися ними;

– визначте, чи конкретні клієнтські сегменти відчувають себе більш чітко про вашу компанію. Деякі демографічні показники, інтереси, персонажі та ін.

Аналіз настроїв корисний для розуміння Голосу Клієнта (VoC), оскільки він допомагає виконувати наступні дії:

– використовуйте результати аналізу настроїв для розробки більш поінформованих питань, щоб задати питання про майбутні дослідження.

– зрозумійте нюанси клієнтського досвіду з часом, а також чому і як відбуваються зміни;

– посилюйте свої внутрішні команди, надаючи їм більш глибокий погляд на досвід клієнтів, на сегменти та на конкретні аспекти бізнесу;

– швидше реагуйте на сигнали та переходи від клієнтів.

Приклад: Проект голосів McKinsey.

McKinsey & Company - міжнародна консалтингова компанія, що спеціалізується на вирішенні завдань, пов'язаних зі стратегічним управлінням. McKinsey як консультант співпрацює з найбільшими світовими компаніями, державними установами і некомерційними організаціями.

У Бразилії федеральні державні витрати зросли на 156% з 2007 по 2015 роки, в той час як задоволеність населення державними послугами постійно зменшувалася. Незадоволений цим контрпродуктивним прогресом, Департамент міського планування набрав McKinsey, щоб допомогти їм працювати над низкою нових проектів, які б зосередилися в першу чергу на досвіді користувачів або подорожах громадян під час надання послуг.

Цей стиль управління, орієнтований на громадян, призвів до того, що ми називаємо Розумні міста.

McKinsey розробила інструмент, який називається City Voices, який проводить опитування громадян (клієнтів) у більш ніж 150 різних показниках, а потім проводить аналіз настроїв, щоб допомогти керівникам зрозуміти, як живуть складові і що їм потрібно, щоб краще інформувати державну політику. Використовуючи цей інструмент, бразильський уряд зміг розв'язати нагальні потреби - наприклад, безпечнішу автобусну систему - і в першу чергу поліпшити їх.

Якщо навіть цілі міста та країни, відомі своєю бюрократією та повільними темпами, включають подорожі клієнтів та аналіз настроїв у свої процеси прийняття рішень, то інноваційні компанії краще будуть далеко попереду.

1.5.4 Аналіз настроїв у службі підтримки клієнтів

Ми всі знаємо, що тренування: зоряний досвід клієнтів = більш ймовірні клієнти, що повертаються. Особливо в останні роки було багато розмов (по праву) навколо досвіду клієнтів та подорожей клієнтів. Провідні компанії почали усвідомлювати, що найчастіше те, як вони поставляють, є такими ж (якщо не більш) важливими, як те, що вони забезпечують. Сьогодні, як ніколи раніше, клієнти очікують, що їхній досвід роботи з компаніями буде негайним, інтуїтивним, особистим та безпроблемним. Фактично, дослідження показують, що 25% клієнтів перейдуть на конкурента після одного негативного взаємодії.

Провідні компанії почали усвідомлювати, що найчастіше те, як вони поставляють, є такими ж (якщо не більш) важливими, як те, що вони забезпечують.

Ми вже подивилися, як можна використовувати аналіз настроїв у розширеному VoC, але тепер ми наберемо спеціальні команди обслуговування клієнтів.

Як можна використовувати аналіз настроїв:

– автоматизація систем для виконання аналізу настроїв на всі вхідні запити щодо підтримки користувачів;

– швидко виявляйте розчарованих клієнтів;

– маршрут запитів до конкретних членів команди найкраще підходить для відповіді;

– використовуйте аналітику, щоб отримати глибоке розуміння того, що відбувається в службі підтримки клієнтів.

Аналіз настроїв корисний у підтримці клієнтів, оскільки допомагає виконувати наступне:

– визначте пріоритетність порядку реагування на квитки, переконавшись, що в першу чергу ви вирішите найбільш нагальні потреби;

– підвищуйте ефективність, автоматично призначаючи квитки певній категорії або члена команди.

Приклад: Аналіз взаємодії з клієнтською підтримкою в Twitter

Було проведено аналіз того, як чотири найбільших американських перевізників телефону (AT&T, Verizon, Sprint, та T-Mobile) обробляли взаємодію підтримки клієнтів у Twitter.

Бцло завантажено десятки тисяч твітів, де були згадані ці компанії (по імені або за допомогою вказівки на користувача), і провели їх через модель MonkeyLearn, щоб класифікувати кожен твіт як позитивний, нейтральний або негативний. Потім ми використали наш новий Insight Extractor, який читає весь текст як одну одиницю, після чого витягує найбільш релевантні ключові слова і повертає найбільш відповідні речення, включаючи кожне окреме ключове слово, що було виявлено.

Ось деякі відомості:

– T-Mobile мав дуже високий відсоток позитивних твітів;

– Verizon була єдиною компанією з більш негативними твітами, ніж позитивні;

– найпопулярніші ключові слова для позитивних твітів у Verizon включали типові терміни, такі як "новий телефон", "спасибі" і "якісне обслуговування клієнтів". Ключові пропозиції були типовими, формальними, трохи сухими взаємодіями між командою та послідовниками;

– найпопулярніші ключові слова для позитивних твітів у T-Mobile включали імена людей у своїй команді підтримки клієнтів, тому що їхня команда практикувала більш високу взаємодію, бесіди з клієнтами були більш персоніфікованими, тому користувачі отримували краще враження про взаємодію з цією компанією.

Підводячи підсумок, це може означати, що більш особистісне, залучення до соціальних медіа викликає більш позитивні відповіді та вищу задоволеність клієнтів. Тому для аналізу настроїв клієнтів доцільно використовувати інструменти аналізу тональностей тексту.

1.5.5 Аналіз настроїв у аналітиці робочої сили та голосу працівника

Так само, як ми вимірюємо VoC через опитування клієнтів, ми можемо вимагати і діяти на основі зворотного зв'язку від наших співробітників. Швидше за все, вони значно більше інвестують у надання дієвих ідей щодо вдосконалення робочого місця. І шанси, що ви, як роботодавець, дико більше зацікавлені в тому, щоб вони були залучені і наділені повноваженнями, щоб зробити все можливе.

Як можна використовувати аналіз настроїв:

– аналізуйте опитування співробітників, вибирайте ключові слова та переглядайте їх за сегментами;

– відстежувати зміни у настроях працівників у часі;

– вирішуйте поверхневі термінові проблеми негайно.

Аналіз настроїв корисний у аналітиці робочих місць і VoE, оскільки допомагає виконувати такі дії:

– відкрийте і вирішуйте проблеми співробітників, гарантуючи, що вони почують і цінують.

– розумійте VoE в реальному часі, а не щорічні огляди або огляди продуктивності.

Припустимо, ви проводите внутрішнє опитування, яке вимагає від співробітників оцінювати різні аспекти свого досвіду на робочому місці і пояснювати, чому вони так вважають. За шкалою від 1 до 10 найуспішніший працівник може сказати, що оцінює свою участь у роботі як 5 - не ідеально. Проте, якщо ми подивимося ближче, то побачимо, що вона додала: «Я люблю роботу, яку я роблю, і мої можливості для навчання були чудовими, але мій бос робить випадкові невідповідні зауваження до мене, які змушують мене відчувати себе незручно»

Відповідь, подібна до цього, повинна підняти червоні прапори потенційних сексуальних домагань і негайно привернути увагу персоналу, щоб вирішити ситуацію. Якщо ви просто скинули його разом з іншими сукупними оцінками і не

читали їх ще два місяці, ви ризикуєте втратити цінного співробітника або підвищити вже напружену ситуацію.

1.5.6 Аналіз настроїв у аналітиці продукту

У нашому гнучкому світі ми дізналися, що продукти краще побудувати на ранніх етапах прототипування, часто вимагаючи зворотного зв'язку і продовжуючи повторювати і вдосконалювати. Але для багатьох виробничих команд, які вимагають частого зворотного зв'язку, може бути найбільш складною частиною. Як зменшити сегмент клієнта, який потрібно задати? Як ви сортуєте і зважуєте всі їхні відгуки? Це саме те, де аналіз настроїв може змінити гру. Незалежно від того, аналізуючи опитування, взаємодії з клієнтами або соціальні медіа, машинне навчання дає змогу одразу оцінити величезну кількість відгуків про продукт.

Як можна використовувати аналіз настроїв:

– проаналізуйте велику кількість досліджень зі зворотним зв'язком з продуктом

– проаналізуйте всі соціальні медіа та онлайн-згадки про продукт

– фільтрувати коментарі по аспектах і по настроях, щоб побачити, що потрібно налаштувати і що потрібно зберегти.

– автоматично направляйте відповідні коментарі до груп продуктів.

Аналіз аналітичних даних корисний для аналітики продуктів, оскільки допомагає виконувати наступні дії:

– слід постійно переходити на вкладки того, що люблять люди та не люблять ваш продукт;

– встановіть ноль, у яких сегментах, які речі, і як звернутися до цих аудиторій;

– допоможіть команді розробників продуктів неймовірне розуміння особливостей продуктивності.

Приклад: MonkeyLearn

Команда проводить аналіз настроїв на взаємодію з клієнтською підтримкою та використовує ці знання для того, щоб розширити можливості кожного в нашій компанії, а не лише для наших агентів підтримки. Тому, коли клієнт згадує, що у них виникають труднощі з X або що вони хочуть бачити Y, ми надаємо цю інформацію безпосередньо людям, які створюють наші продукти та керують ними. Ми мали реальний зворотний зв'язок з реальними клієнтами, безпосередньо досягаючи вух людей, до яких це мало найбільше. Як і будь-яка велика команда виробників, ми слухаємо клієнтів і задовольняємо їхні потреби. Занадто часто все, що потрібно, - це просто оснащення вашої команди правильним розумінням, що співпадає з клієнтом.

1.5.7 Аналіз настроїв у дослідженні та аналізі ринку

І в якості остаточного випадку використання, аналіз настроїв надає переваги усім видам маркетингових досліджень і конкурентного аналізу. Незалежно від того, чи вивчаєте ви новий ринок, передбачаєте майбутні тенденції або зберігаєте перевагу в конкуренції, аналіз настроїв може зробити всі зміни.

Як можна використовувати аналіз настроїв:

– проаналізуйте огляди продуктів вашого бренду та порівняйте їх з конкурентами4

– створюйте щотижневі, щомісячні або щоденні звіти - своєрідна система раннього попередження;

– порівняйте настрої на міжнародних ринках;

– проаналізуйте офіційні звіти ринку або бізнес-журнали для довгострокових, більш широких тенденцій;

– аналізуйте твіти та повідомлення соціальних медіа для подій у реальному часі;

– проаналізуйте відгуки про нефільтровані відгуки клієнтів;

– використовуйте аналіз на основі аспектів, що базується на аспектах, щоб отримати багате уявлення про деталі та причину непрозорих тенденцій на ринку.

Аналіз настроїв корисний для маркетингових досліджень та аналізу, оскільки допомагає:

– доторкніться до нових джерел інформації.

– кількісно визначити якісну інформацію.

– додайте цей якісний вимір до вже зібраних кількісних уявлень.

– надайте інформацію в реальному часі, а не в ретроспективі.

– автоматизація для регулярних (можливо, щотижневих) звітів.

– заповніть прогалини, де громадські дані є дефіцитними - наприклад, на ринках, що розвиваються.

Приклади: Відгуки про готелі на TripAdvisor

Наша команда цікавилася тим, як люди відчувають себе в готелях у кількох великих містах світу, тому ми зібрали та проаналізували більше мільйона відгуків від TripAdvisor. Ми дивилися на готелі в Лондоні, Парижі, Нью-Йорку, Бангкоку, Мадриді, Пекіні та Ріо-де-Жанейро. В основному, відгуки були позитивними - у середньому 82% речей, які написали люди, позначені позитивним настроєм. Але готелі Лондона отримали найгірші відгуки. Лондон розглядався як більш брудний, ніж Нью-Йорк і з найгіршою їжею в цілому.

1.6 Постановка задачі

Метою роботи є дослідження існуючих методів аналізу тональності текстів.

Враховуючи сучасні можливості нейронних мереж та машинного навчання, було прийнято рішення досліджувати методи аналізу тональностей тексту з використанням нейронних мереж. Отже, буде проведено дослідження можливостей використання різних типів архітектур та підходів у роботі з нейронними мережами для задачі аналізу тональностей текстів.

Дослідження буде проведено на наборі даних IMDB Movie Reviews Dataset[10]. Цей набір даних містить 50000 відгуків на фільми з сайту Internet Movie Database - це найбільша в світі база даних та веб-сайт про кінематограф, який містить не лише вичерпну інформацію про фільми, серіали та ін., а також дає можливість користувачам ставити оцінки фільмам та писати до них відгуки. Саме ці відгуки буде використано для проведення дослідження. Набір даних розділено на два набори: 25000 для тренувального набору та 25000 для тестувального набору. Відгуки у даному наборі розділені лише на два типи: негативні та позитивні (набір містить рівну кількість позитивних та негативних відгуків), тобто буде проводитися бінарна класифікація.

Для проведення дослідження були обрані такі класи нейронних мереж та підходи:

– згорткова нейронна мережа (convolutional neural network, CNN);

– довга короткочасна пам’ять (long short-term memory, LSTM), варіант архітектури рекурентної нейронної мережі;

– використання попередньо навченої моделі;

Результатом проведеного дослідження буде порівняння точності класифікації відгуків як позитивних чи негативних для трьох зазначених вище підходів. Мірою точності буде виступати відсоток вірних передбачень з 25 тисяч відгуків.

Також буде розроблено веб-додаток, у якому користувач матиме можливість визначити емоціональне забарвлення довільного тексту, використовуючи три варіанти попередньо тренованих нейронних мереж: згорткова нейронна мережа, рекурентна нейронна мережа (архітектура коротка довгочасна пам’ять), та модель Google’s BERT (опис кожного підходу буде наведено нижче.

2 АНАЛІЗ МЕТОДІВ ДЛЯ ДОСЛІДЖЕННЯ

2.1 Огляд існуючих датасетів та рішень

2.1.1 Огляд датасетів

Ключовою частиною для освоєння аналізу настроїв є робота над різними наборами даних та експериментальні різні підходи. Для цього спочатку потрібно отримати дані та отримати набір даних, над яким ви будете робити свої експерименти на основі вашого домену та інтересів.

Нижче наведено деякі з найбільш популярних наборів даних для аналізу експериментів з аналізом настроїв та підходу до машинного навчання. Вони відкриті та безкоштовні для завантаження[5].

Відгуки про продукт: цей набір даних складається з кількох мільйонів відгуків клієнтів Amazon зі зірковими рейтингами, надзвичайно корисних для навчання моделі аналізу настроїв.

Відгуки ресторанів: цей набір даних складається з 5,2 мільйонів відгуків Yelp з рейтингом зірок.

Огляди фільмів: цей набір даних складається з 1000 позитивних і 1000 негативних оброблених відгуків. Він також надає 5,331 позитивних і 5,331 негативних оброблених пропозицій / фрагментів.

Точні відгуки про продукти харчування: цей набір даних складається з ~ 500 000 відгуків про продукти харчування від Amazon. Вона включає в себе інформацію про продукт і користувача, рейтинги та звичайну текстову версію кожного огляду.

Відгуки авіакомпанії Twitter на Kaggle: цей набір складається з ~ 15000 позначених твітів (позитивний, нейтральний і негативний) про авіакомпанії.

Перший дебат у групі GOP Twitter Sentiment: цей набір даних складається з ~ 14 000 позначених твітів (позитивних, нейтральних та негативних) щодо першої дискусії GOP у 2016 році.

Якщо ви зацікавлені в підході, що ґрунтується на правилах, нижче наведено різноманітний список лексиконів аналізу настроїв, які будуть корисними. Ці

лексикони забезпечують набір словників слів з мітками, що визначають їхні настрої в різних областях[4]. Наступні лексикони дійсно корисні для визначення настрою текстів:

Лексикони настроїв для 81 Мови: цей набір містить як позитивні, так і негативні лексики настроїв для 81 мови.

SentiWordNet: цей набір даних містить близько 29 000 слів із оцінкою настроїв від 0 до 1.

Лексикон думки для аналізу настроїв: цей набір даних містить список з 4782 негативних слів і 20000 позитивних слів англійською мовою.

Слово словника Wordstat: цей набір даних містить ~ 4800 позитивних і ~ 9000 негативних слів.

Emoticon Sentiment Lexicon: цей набір даних містить список 477 смайликів, позначених як позитивні, нейтральні або негативні.

Лексикон AFINN є, мабуть, одним з найпростіших і найпопулярніших лексиконів, які можна широко використовувати для аналізу настроїв. Більш детальну і повну інформацію про цю лексику, розроблену та опрацьовану Фінном Орупом Нільсен, можна знайти у статті «Новий ANEW: оцінка списку слів для аналізу настроїв у мікроблогах». Поточна версія лексикону - AFINN-en-165. txt і містить більше трьох тисяч трьох ста слів з оцінкою полярності, пов'язаної з кожним конкретним словом. Цей лексикон у вільному доступі можна знайти на офіційному сховищі GitHub автора разом з попередніми версіями його, включаючи AFINN-111.

2.1.2 API для аналізу настроїв

Існує декілька варіантів систем аналізу настроїв, які можна використовувати через API. Загалом, їх можна розділити на дві різні категорії: бібліотеки з відкритим кодом та комерційні рішення.

Python є однією з провідних мов програмування для науки про дані, і вона має сильну спільноту і великий набір варіантів для реалізації моделей NLP.

Scikit-learn - це бібліотека для машинного навчання і корисні інструменти для векторизації тексту. Навчання класифікатору поверх векторизацій, таких як частота або векторизатори тексту tf-idf, дуже проста. Scikit-learn має втілення для підтримки векторних машин, наївних байєсів і логістичної регресії.

NLTK - традиційна бібліотека для Python. Вона має активне співтовариство, і, крім того, що забезпечує низький рівень функцій для НЛП, вона також надає можливість тренувати класифікатори машинного навчання.

SpaCy є ще однією новою бібліотекою НЛП зі зростаючою спільнотою. Як і NLTK, він забезпечує сильний набір функцій низького рівня для НЛП і підтримку для підготовки класифікаторів тексту.

Завдяки тенденції Deep Learning в останні кілька років було розроблено новий набір бібліотек з інформатики, які підтримують застосування NLP.

TensorFlow. Розроблений Google, він надає низькорівневий набір інструментів для створення та навчання нейронних мереж. Також існує підтримка векторизації тексту, як на традиційній частоті слова, так і на більш просунутих за допомогою вбудовування слів.

Keras надає корисні абстракції для роботи з декількома типами нейронних мереж, такими як рекурентні нейронні мережі (RNNs) і згорткові нейронні мережі (CNNs) і легко укладають шари нейронів[11]. Керас можна керувати на вершині Tensorflow або Theano. Він також надає корисні інструменти для класифікації тексту.

PyTorch - це новітня система глибокого навчання, яку підтримують деякі престижні організації, такі як Facebook, Twitter, Nvidia, Salesforce, Стенфордський університет, Оксфордський університет і Uber. Він швидко розвинув сильну громаду.

TextBlob - це ще одна чудова з відкритим кодом для виконання завдань НЛП з легкістю, включаючи аналіз настроїв. Це також лексикон настрою (у вигляді XML-файлу), який він використовує, щоб дати оцінки полярності та суб'єктивності.

Як правило, оцінки мають нормалізований масштаб порівняно з Afinn. Оцінка полярності являє собою поплавок у діапазоні [-1.0, 1.0]. Суб'єктивність є плаваючою в діапазоні [0.0, 1.0], де 0.0 дуже об'єктивно і 1.0 дуже суб'єктивно. Давайте скористаємося цим зараз, щоб отримати полярність і мітки настроїв для кожної статті новин і об'єднати підсумкові статистичні дані для кожної категорії новин.

2.2 Аналіз обраних для дослідження методів

2.1.2 Згорткові нейронні мережі

Згорткова нейронна мережа (convolutional neural network, CNN) – це класс нейронних мереж глубокого навчання, що зазвичай використовується для аналізу зображень, відео, а також обробки природних мов[12]. Згорткова нейронна мережа є регуляризованною версією багатошарового персептрона, який розроблено таким чином, щоб для роботи нейронної мережі потрібно було проводити мінімальну попередню обробку.

Робота згорткової нейронної мережі зазвичай є переходом від конкретних особливестей вхідних даних до більш абстрактних деталей, і далі до ще більш абстрактних деталей, доходячи до виділення понять високого рівня. Мережа є самоналаштованною і самостійно виробляє необхідну ієрархію абстрактних ознак чи послідовностей ознак, проводячи фільтрацію неважливих деталей і виділяючи важливі.

Ознаки, які виробляє нейронна мережа, зазчивай є доволі складними для розуміння, тому у випадку, якщо система ігнорує якісь істотні ознаки, замість зміни змісту ознак рекомендується удосконалити структуру та архітектуру мережі.

У звичайному перцептроні, який представляє собою повнозв'язну нейронну мережу, кожен нейрон пов'язаний з усіма нейронами попереднього шару, причому кожна зв'язок має свій персональний ваговий коефіцієнт[12].

У згортковій нейронній мережі в операції згортки використовується лише обмежена матриця ваг невеликого розміру, яку «рухаються» по всьому оброблюваного шару (на самому початку - безпосередньо по вхідним даним), де формують після кожного зсуву сигнал активації для нейрона наступного шару з аналогічною позицією (див. рис. 2.1).

Тобто для різних нейронів вихідного шару використовуються одна і та ж сама матриця ваг, яку також часто називають ядром згортки. Її інтерпретують як графічне кодування якої-небудь ознаки, наприклад, наявність похилої лінії під певним кутом чи наявність певних фігур, що наприклад повторюються на різних зображеннях. Тоді наступний шар, що вийшов в результаті операції згортки такою матрицею ваг, показує наявність даної ознаки в оброблюваному шарі і її координати, формуючи так звану карту ознак (англ. Feature map). При цьому такі ядра згортки не закладаються дослідником заздалегідь, а завжди формуються самостійно шляхом навчання мережі класичним методом зворотного поширення помилки.

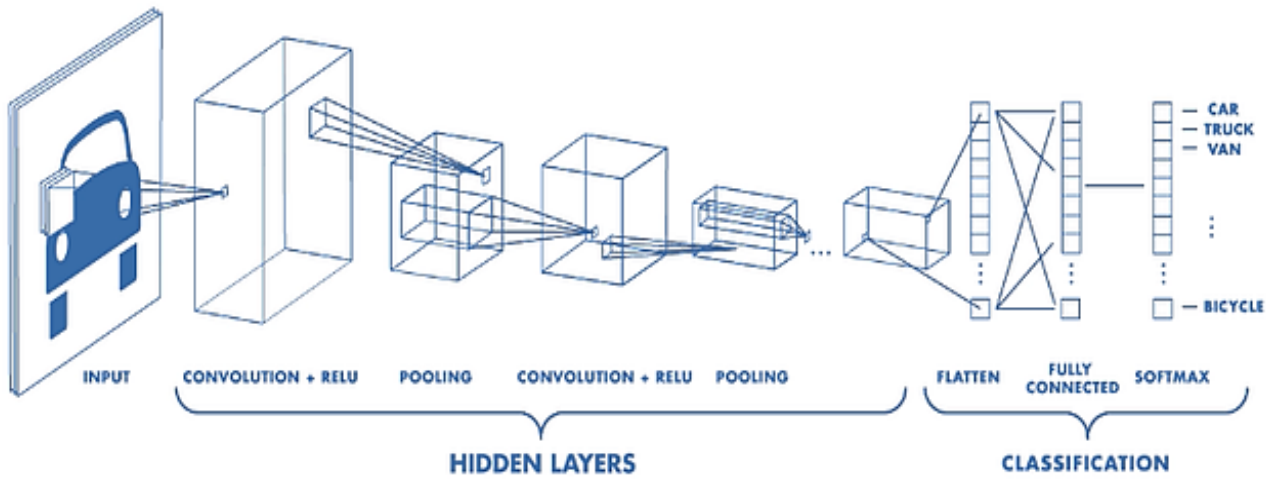


Рисунок 2.1 - Типова архітектура згорткової нейронної мережі

Природно, в згортковій нейронній мережі набір ваг не один, а ціла гама, що кодує елементи вхідних даних (наприклад лінії і дуги під різними кутами для зображення).

При цьому такі ядра згортки не закладаються дослідником заздалегідь, а формуються самостійно шляхом навчання мережі класичним методом зворотного поширення помилки.

Прохід кожним набором ваг формує свій власний примірник карти ознак, роблячи нейронну мережу багатоканальної (багато незалежних карт ознак на одному шарі). Також слід зазначити, що при переборі шару матрицею ваг її пересувають зазвичай не на повний крок (розмір цієї матриці), а на невелику відстань. Так, наприклад, при розмірності матриці ваг 5 × 5 її зрушують на один або два нейрона (пікселя) замість п'яти, щоб не «переступити» шукану ознаку.

Найбільш простим і популярним способом навчання є метод навчання з учителем (на маркованих даних) - метод зворотного поширення помилки і його модифікації.

Але існує також ряд технік навчання згортовноъ мережі без вчителя. Наприклад, фільтри операції згортки можна навчити окремо і автономно, подаючи на них вирізані випадковим чином шматочки вихідних зображень навчальної вибірки і застосовуючи для них будь-який відомий алгоритм навчання без вчителя (наприклад, автоассоціатор або навіть метод k-середніх) - така техніка відома під назвою patch-based training.

Відповідно, наступний шар згортки мережі буде навчатися на шматочках від уже навченого першого шару мережі. Також можна скомбінувати сверточное нейросеть з іншими технологіями глибинного навчання. Наприклад, зробити згортовий авто-асоціатор, згорткову версію каскадних обмежених машин Больцмана, що навчаються за рахунок імовірнісного математичного апарату, згорткову версію розрідженого кодування (англ. Sparse coding), названу deconvolutional networks («розгорткові» мережі)[13].

Для проведення дослідження роботи згорткової нейронної марежі буде використовуватися Keras. Це відкрита бібліотека для роботи з нейронними мережами, яка містить необхідні методи для обробки вхідних даних у необхідний для роботи мережі вигляд.

2.2.2 Рекурентна нейронна мережа - довга та короткочасна пам’ять

Рекурентна нейронна мережа (Recurrent neural network; RNN) – клас нейронних мереж, в якому міжвузлові з’єднання утворюють направлену у часі послідовність (орієнтований граф)[14]. Завдяки цьому рекурентні нейронні мережі надають можливість опрацьовувати послідовні просторові ланцюги чи серії подій у часі.

Рекурентні мережі, на відміну від багатошарових перцептронів, можуть використовувати свою внутрішню пам’ять для обробки послідовностей довільної довжини. Через це подібні нейронні мережі є широко застосовними в задачах розпізнавання тексту чи мови, а також для задач обробки природних мов[15], де вони дозволяють широко використовувати попередній досвід.

Найбільш розповсюдженими є варіанти архітектури рекурентної нейронної мережі з довгою короткочасною пам’яттю (LSTM) та керованим рекурентним блоком (GRU).

У дослідженні буде використано варіант архітектури з довгою короткочасною пам’яттю.

LSTM-мережа добре пристосована для задач з класифікації[16]. Ця штучна нейронна мережа містить LSTM-модулі замість або в якості доповнення до інших мережевих модулів.

LSTM-модуль це рекурентний модуль мережі, який здатен запам’ятовувати значення як на короткі, так і на довгі проміжки часу. LSTM-модуль не використовує функцію активації в середині своїх рекурентних модулів, і значення, що зберігається, не розмивається у часі, тому градієнт не зникає, коли використовується метод зворотного розпізнавання розповсюдження помилки у часі під час тренування мережі.

LSTM-модулі часто складаються у блоки. Таке будування характерно для глубоких багатошарових нейронних мереж і дозволяє використовувати паралельні обчислення та спеціальне обладнання.

LSTM-блоки містять три або чотири “вентилі” (gates), що використовуються для контролю інформації на входах та виходах пам’яті цих блоків (рис. 2.2).

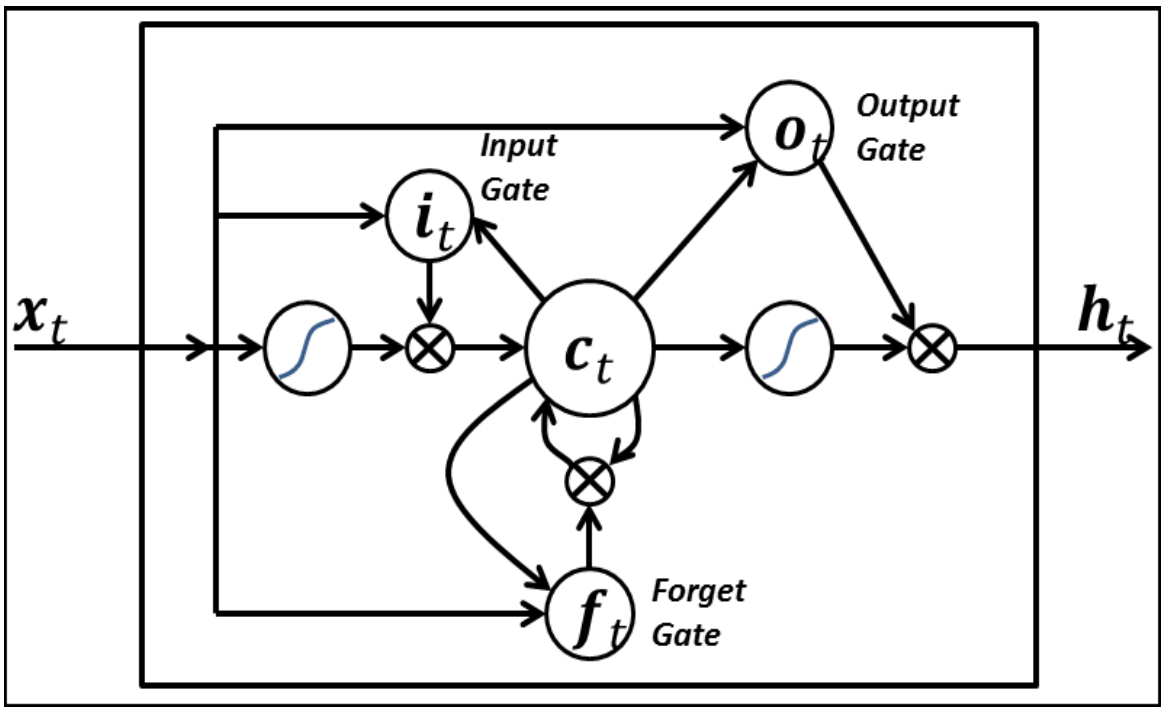


Рисунок 2.2 - LSTM-блок з трьома вентилями

Ці вентилі реалізовані у вигляді логістичної функції для обчислення значень в діапазоні [0; 1]. Множення на це значення використовується для часткового допуску чи заборони потока інформації в середину або назовні пам’яті[17]. Наприклад, «вхідний вентиль» (input gate) керує мірою входження нового значення у пам’ять, «вентиль забуття» (forget gate) керує тим, до якої міри значення залишається в пам'яті. А «вихідний вентиль» (англ. output gate) керує тим, до якої міри значення в пам'яті використовується для обчислення активації виходу блоку.

Ваги в LSTM-блоці (W і U) використовуються для завдання напрямку оперування вентилів. Ці ваги визначені для значень, які подаються в блок (x_t і вихід з попереднього тимчасового кроку h_{t-1}) для кожного з вентилів. Таким чином, LSTM-блок визначає, як розпоряджатися своєю пам'яттю як функцією цих значень, і тренування ваг дозволяє LSTM-блоку вивчити функцію, що мінімізувала втрати. LSTM-блоки зазвичай тренують за допомогою методу зворотного поширення помилки в часі.

LSTM-мережі застосовують для керування роботами, розпізнавання мовлення, навчання граматики, розпізнавання дій людей, прогнозування часових рядів, навчання ритму, виявлення гомології білків, навчання ритму[18].

2.2.3 Попередньо навчені моделі нейронних мереж

Попередньо підготовлена модель – це збережена мережа, яка раніше навчалася на великому наборі даних, як правило, на масштабному завданні класифікації зображень. У цьому підході або проходить використання моделі, яка вже була оброблена, або використовується трансферне навчання, щоб налаштувати цю модель для даного завдання.

Трансферне навчання - це методика машинного навчання, де модель, що навчається на одному завданні, переорієнтована на друге відповідне завдання. Згідно з означенням з книги Deep Learning[19], трансферне навчання та адаптація домену відносяться до ситуації, коли те, що було вивчено в одній установці використовується для поліпшення узагальнення в іншому середовищі. Це вдосконалення навчання в новому завданні через передачу знань з пов'язаного завдання, яке вже було вивчено.

Ідея трансферного навчання полягає в тому, що якщо модель навчена на великому і загальному наборі даних, то ця модель буде ефективно служити загальною моделлю візуального світу. Потім є можливість скористатися цими вивченими картами властивостей без необхідності починати з нуля навчання великої моделі на великому наборі даних.

В останні роки глибоке навчання досягло значного прогресу. Це дозволило нам вирішувати складні проблеми і давати дивовижні результати. Однак час навчання та обсяг даних, необхідних для таких систем глибокого навчання, набагато більше, ніж у традиційних систем ML. Існують різні глибокі мережі навчання з найсучаснішими діями (іноді так добре або навіть краще, ніж

продуктивність людини), які були розроблені та випробувані в різних областях, таких як комп'ютерне бачення та обробка природної мови[20]. У більшості випадків команди/люди діляться деталями цих мереж, щоб інші могли їх використовувати. Ці попередньо навчені мережі/моделі формують основу трансферного навчання в контексті глибокого навчання, або те, що можна називати "глибоким трансферним навчанням".

Для проведення аналізу тональностей тексту можуть використовуватися багатоцільові моделі для обробки природної мови. Ці моделі підсилюють застосування для обробки природних мов - машинний переклад, системи відповіді на запитання, чат-боти, аналіз настроїв і т.д. Основний компонент цих багатоцільових моделей обробки природних мов – концепція мовного моделювання.

Найвідомішими є наступні попередньо треновані моделі:

– ULMFiT – модель, натренована на наборі даних з сайту Wikitext, завдяки чому модель може використовуватися для широкого спектру задач обробки природної мови;

– Transformer – модель, представлена компанією Google, для тренування якої були використані згорткові та рекурентні нейронні мережі;

– Google’s BERT – інша попередньо натренована модель від Google, яка містить результати виконання 11 задач обробки природної мови, в тому числі аналіз тональностей тексту[21];

– ELMo – модель, що використовує вбудовування слів (англ. Word Embeddings) – перетворення текстових даних у числові.

Для проведення дослідження буде використовуватися модель Google’s BERT.

BERT – це метод попередньої підготовки мови, що означає, що тренується загальна модель "розуміння мови" на великому текстовому корпусі (наприклад, з Вікіпедії), а потім ця модель використовується для задач обробки природної мови,. BERT перевершує попередні методи, оскільки це перша без нагляду, глибоко двонаправлена система для попереднього навчання обробки природної мови [21].

У данному контексті характеристика «Без нагляду означає», що BERT навчався, використовуючи лише звичайний текст, що є важливим, оскільки величезна кількість звичайних текстових даних є загальнодоступною в Інтернеті багатьма мовами.

Попередньо навчені представлення також можуть бути контекстно-вільними або контекстуальними, а контекстні подання можуть бути однонаправленими або двонаправленими. Контекстні моделі, такі як word2vec або GloVe, створюють єдине "вбудовування слова" для кожного слова у словнику, тому банк матиме таке ж уявлення в банківському депозиті та річковому березі. Контекстні моделі замість цього створюють представлення кожного слова, яке базується на інших словах у реченні.

3 ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ

3.1 Інструменти та дані для дослідження

Для проведення дослідження було обрано набір даних, що містить 50000 відгуків на фільми з бази IMDB. Відгуки в цьому наборі даних промарковані як позитивні чи негативні, що дозволяє проводити бінарний аналіз тональності текстів. Відгуки написані англійською мовою.

Приклад позитивного відгуку до фільму “Титанік”:

“Why do people bitch about this movie and not about awful movies like The Godfather. Titanic is the greatest movie of the 21st Century.With great acting,directing,effects,music and generally everything. This movie is always dumped by all because one day some one said they didn't like it any more so most of the world decided to agree. There is nothing wrong with this movie. All I can say is that this movie, not only being the most heavily Oscar Awarded movie of all time, the most money ever made ever and sadly one of the most underrated movies I've ever seen. Apart from that it is truly the best movie of all time. The only movies that come close to being like all the Star Wars and the Lord of the Rings trilogy or anything by the masters Hitchcock or Spielberg or Tim Burton. These are all good movies and directors but none match up to James Cameron's Masterpiece TITANIC.”

Приклад негативного відгуку до фільму “Титанік”:

“1st watched 5/17/2002 - 3 out of 10(Dir-Ewald Andre Dupont): Fairly lame account of the Titanic disaster is the first filmed version of this much-heralded event. The replication of the disaster is not bad, but the drama around it is at some times silly, badly acted and way-too soap opera-like. The story is very much the same as the most recent Oscar-winning one except that we are shown how the crew tried to hide the actual disaster that was occurring until almost too late. Good for nostalgia purposes only and to get a feel for what James Cameron was competing against(barelyÂ…) in his recreation.”

Відгуки в наборі поділені на дві рівні частини: дані для тренування (25000 відгуків) та дані для тестування (25000 відгуків). Обидва набори містять рівну

кількість позитивних та негативних відгуків (рис. 3.1). Також набір даних додатково містить 50000 відгуків без інформації про забарвлення тексту.

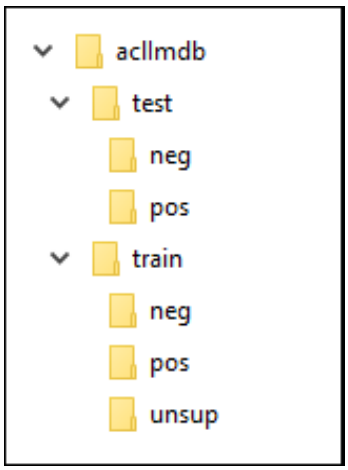


Рисунок 3.1 – Структура набору даних

У якості основної мови розробки для проведення дослідження було обрано мову програмування Python.

Python є інтерпретованою мовою програмування високого рівня загального призначення. Створений Гвідо ван Россумом і вперше випущений в 1991 році, Його мовні конструкції та об'єктно-орієнтований підхід спрямовані на те, щоб допомогти програмістам написати чіткий, логічний код для малих і великих проектів.

Наразі пайтон широко використовується для виконання задач машинного навчання та наукових розрахунків. Простота синтаксису та відносно невисокий поріг входу роблять його дуже привабливим для проведення досліджень як без, так і з використанням машинного навчання та нейронних мереж. Пайтон є кросс-платформенним, тому може використовуватися під різними операційними системами, під нього створені дуже зручні середовища розробки, як бескоштовні, так і комерційні. Окрім цього, для мінімального використання Пайтону не потрібні навіть середовища розробки.

Але найбільшою перевагою Пайтону є велика кількість бібліотек, що можуть бути використані для наукових досліджень з використанням машинного навчання та нейронних мереж. Використані в дослідженні бібліотеки та платформи буде описано нижче.

Для роботи з згортковими нейронними мережами буде використовуватися TensorFlow. TensorFlow - це відкрита платформа з відкритим вихідним кодом для машинного навчання. Вона має всеосяжну, гнучку екосистему інструментів, бібліотек та ресурсів спільноти, що дозволяє дослідникам підштовхувати сучасні технології розробки ML, а розробники легко створюють та розгортають додатки, що працюють на основі ML.

Для роботи з TensorFlow буде використовуватися Keras. Keras - це високоякісний API для створення моделей глибокого навчання[15]. Він використовується для швидкого створення прототипів, складних досліджень, а також для створення додатків. Три ключових переваги Keras API:

– простота в використанні: Keras має простий інтерфейс, оптимізований для більшості розповсюджених задач глубокого навчання. Також він дає конкретні підказки як швидко виправити можливі помилки;

– модульність: моделі Keras будуються за допомогою об'єднання декількох простих модулів, кожен з яких може бути налаштований незалежним чином;

– легко розширити модель: користувач може створювати власні модулі, необхідні для конкретного дослідження.

Для роботи з датасетом буде використовуватися бібліотека NumPy. NumPy є основним пакетом для наукових обчислень з Python. Він містить, серед іншого:

– потужний об'єкт N-розмірного масиву;

– складні широкоформатні функції;

– інструменти для інтеграції C / C ++ і Fortran;

– корисна лінійна алгебра, перетворення Фур'є і можливості випадкових чисел.

Крім очевидного наукового використання, NumPy також може використовуватися як ефективний багатовимірний контейнер загальних даних. Можуть бути визначені довільні типи даних. Це дозволяє NumPy легко і швидко інтегруватися з широким спектром баз даних.

Для роботи з числовими таблицями буде використовуватися бібліотека Pandas. Це програмна бібліотека мови Python для обробки та аналізу даних. Работа

панд з даними будується поверх бібліотеки NumPy, що є інструментом більш низького рівня. При наявності пакету matplotlib ця бібліотека дає можливість малювати графіки на отриманих наборах даних.

Для роботи з LTSM та попередньо тренованою моделлю буде використовуватися PyTorch. Це бібліотека машинного навчання для мови Python з відкритим вихідним кодом, створена на базі Torch[18]. Пакет torch реалізує основну структуру даних бібліотеки – n-мірний тензор, а також базові методи для роботи з ним – стандартні математичні та статистичні операції, базові підпрограми лінійної алгебри. Використовується для обробки природної мови.

Для реалізації користувацького інтерфейсу використовується бібліотека React, одна з сучасних JavaScript-бібліотек з відкритим вихідним кодом для розробки користувацьких інтерфейсів.

Основним середовищем розробки обрано PyCharm від JetBrains. Це інтегроване середовище розробки для мови програмування Python. Надає засоби для аналізу коду, графічний відладчик, інструмент для запуску юніт-тестів і підтримує веб-розробку на Django.

У якості платформи використовується Windows.

3.2 Дослідження ефективності застосування згорткових нейронних мереж

Keras має вбудований доступ до датасету відгуків з сайту IMDB і може бути завантажений єдиною командою keras.datasets.imdb.load_data(). Формат датасету вже готовий для використання у нейронних мережах, тому можна пропустити етапи з конвертацією у lower case та видаленням знаків пунктуації. Щоб почати використовувати датасет, треба лише правильно вказати його назву та викликати вбудовану функцію Keras.

Навчання нейронної мережі проводиться за наступним алгоритмом:

– підраховується кількість використань кожного слова у датасеті;

– будується словник використаних слів, де ключом виступає порядковий номер слова, якщо відсортувати масив слів за кількістю використань; так, найбільш використаним є слово “this”;

– кожний відгук приводиться до довжини в 256 слів;

– кожний відгук конвертується в тензор;

– будується модель нейронної мережі;

– проводиться навчання нейронної мережі на тренувальному наборі даних;

– перевіряється результат роботи моделі на тестових даних.

Розглянемо детальніше етапи, починаючи з приведення до довжини.

Конвертація відгуків в тензори може бути зроблена декількома способами.

One-hot encoding конвертує масиви в вектори 0 і 1. Наприклад, послідовність [3, 5] стане 10,000-мірним вектором, повністю складається з нулів крім показників 3 і 5, які будуть представлені одиницями. Потім, нам потрібно буде створити перший Dense шар в нашій мережі, який зможе приймати вектор дані з плаваючою комою. Такий підхід дуже вимогливий до обсягу пам'яті, незважаючи на те, що вимагає вказати розміри матриці num_words * num_reviews.

Інший спосіб - зробити все масиви однаковими по довжині, а потім створити тензор цілих чисел із зазначенням max_length * num_reviews. Ми можемо використовувати Embedding (пер. "Вбудований") шар, який може використовувати ці параметри в якості першого шару нашої мережі.

Обрано було саме другий варіант, бо він менш вибагливий до обсягу пам’яті. Тому першим етапом проведено приведення відгуків до загальної довжини в 256 слів. Це можна зробити за допомогою вбудованої функції pad_sequences. Одразу обрізаємо і тренувальні, і тестові відгуки. Зразок програмного коду для здійснення цієї операції надано нижче.

```
    train_data =
keras.preprocessing.sequence.pad_sequences(train_data,
value=word_index["<PAD>"],
padding='post',
maxlen=256)
    test_data =
keras.preprocessing.sequence.pad_sequences(test_data,
```

```
value=word_index["<PAD>"],
padding='post',
maxlen=256)
```

Наступний етап – власне побудування моделі.

Для створення класифікатора всі верстви проходять процес стека, або накладення:

– перший Embedding шар приймає перекладені в цілі числа слова і шукає відповідний вектор для кожної пари слово/число. Модель навчається на цих векторах. Вектори збільшують розмір одержуваного масиву на 1, в результаті чого ми отримуємо вимірювання: (batch, sequence, embedding);

– наступний шар GlobalAveragePooling1D повертає отриманий вектор заданої довжини для кожного прикладу, усереднюючи розмір ряду. Це дозволить моделі легко приймати дані різної довжини;

– цей вектор пропускається через повнозв'язний Dense шар з 16 прихованими блоками;

– останній шар також є повнозв'язним, але з усього одним вихідним вузлом. За допомогою функції активації sigmoid (сигмоид) отримується число з плаваючою комою між 0 і 1, яке буде показувати ймовірність або впевненість моделі.

Нижче наведено приклад програмного коду для побудови моделі нейронної мережі:

```
    vocab_size = 25000
    model = keras.Sequential()
    model.add(keras.layers.Embedding(vocab_size, 16,
input_shape=(None,)))
    model.add(
    keras.layers.GlobalAveragePooling1D())
    model.add(keras.layers.Dense(16,
activation=tf.nn.relu))
    model.add(keras.layers.Dense(1,
activation=tf.nn.sigmoid))
    model.summary()
```

Вищеописана модель має 2 проміжних або приховані прошарки, між входом і виходом даних. Кількість виходів (блоків, нодів або нейронів) є розміром

репрезентативного простору шару. Іншими словами, кількість свободи, яка дозволена мережі під час навчання.

Якщо модель має більше прихованих блоків, і/або більше шарів, то тоді нейросеть може навчитися більш складним уявленням. Однак в цьому випадку це буде дорожче з точки зору обчислювальних ресурсів і може призвести до навчання небажаних патернів – патернів, які покращують показники на тренувальних даних, але не на перевірочних. Це називається перенавчанням.

Для моделі необхідно вказати функцію втрат і оптимізатор для навчання. Оскільки розв'язувана задача є прикладом бінарної класифікації та модель буде показувати ймовірність (шар з єдиного блоку з сигмоид як функції активації), то буде використовуватися функція втрат binary_crossentropy (пер. "Перехресна ентропія").

Це не єдиний вибір для функції втрат: можна, наприклад, вибрати mean_squared_error. Але зазвичай binary_crossentropy краще справляється з вірогідністю – вона вимірює "дистанцію" між розподілами ймовірностей, або, як у нашому випадку, між еталоном і прогнозами.

При подальшому налаштуванні моделі використовується оптимізатор Адама і перехресна ентропія для втрат:

```
model.compile(
              optimizer=tf.train.AdamOptimizer(),
              loss='binary_crossentropy',
              metrics=['accuracy'])
```

Модель налаштована, і далі проводиться тренування моделі на тренувальній частині датасету, яка містить двадцять п'ять тисяч відгуків, порівну негативних та позитивних.

Тренування моделі починається з 40 епох за допомогою міні-батчів по 512 зразків (Батч - набір, пакет даних). Це означає, зроблено 40 ітерацій (або проходів) по всім зразкам даних в тензори x_train і y_train (де x_train це власне відгуки до фільмів, представлені у вигляді векторів чисел, а y_train це вказання на те, відгук є позитивним чи негативним). Якщо модель має більше прихованих блоків, і/або більше шарів, то тоді нейросеть може навчитися більш складним уявленням.

Результати проходження начання для перших десяти епох зображено на рисунку 3.2.

```
Epoch 1/40
15000/15000 [==============================] - 1s 81us/sample - loss: 0.6910 - acc: 0.6036 - val_loss: 0.6879 - val_acc: 0.6869
Epoch 2/40
15000/15000 [==============================] - 1s 68us/sample - loss: 0.6827 - acc: 0.7347 - val_loss: 0.6771 - val_acc: 0.7554
Epoch 3/40
15000/15000 [==============================] - 1s 72us/sample - loss: 0.6663 - acc: 0.7686 - val_loss: 0.6570 - val_acc: 0.7599
Epoch 4/40
15000/15000 [==============================] - 1s 70us/sample - loss: 0.6392 - acc: 0.7748 - val_loss: 0.6271 - val_acc: 0.7613
Epoch 5/40
15000/15000 [==============================] - 1s 69us/sample - loss: 0.6015 - acc: 0.8028 - val_loss: 0.5888 - val_acc: 0.7917
Epoch 6/40
15000/15000 [==============================] - 1s 67us/sample - loss: 0.5559 - acc: 0.8211 - val_loss: 0.5458 - val_acc: 0.8112
Epoch 7/40
15000/15000 [==============================] - 1s 70us/sample - loss: 0.5065 - acc: 0.8375 - val_loss: 0.5005 - val_acc: 0.8262
Epoch 8/40
15000/15000 [==============================] - 1s 69us/sample - loss: 0.4586 - acc: 0.8549 - val_loss: 0.4600 - val_acc: 0.8392
Epoch 9/40
15000/15000 [==============================] - 1s 71us/sample - loss: 0.4152 - acc: 0.8691 - val_loss: 0.4246 - val_acc: 0.8488
Epoch 10/40
15000/15000 [==============================] - 1s 69us/sample - loss: 0.3778 - acc: 0.8793 - val_loss: 0.3960 - val_acc: 0.8555
```

Рисунок 3.2 – Результати тренування нейронної мережі

Після навчання вимірюються втрати і точність нашої моделі шляхом перевірки на 25,000 зразків з перевірочного набору даних. Для перевірки моделі використовується функція evaluate, що отримує на вхід два масиви – тестові рецензії та відповідні маркування позитивної/негативної рецензії.

Результатом виконання цієї функції є пара чисел: відсоток втрат (loss; чим нижче це число, тим менше хибних прогнозів зробила нейронна мережа) та точність accuracy (див. рис. 3.3).

```
>>> print(results)
25000/25000 [==============================] - 1s 55us/sample - loss: 0.3346 - acc: 0.8699
[0.3346132341432, 0.8699]
```

Рисунок 3.3 – Результати роботи нейронної мережі на тестових даних.

Для більш наглядного відображення результатів побудовано графік втрат та точності для обох етапів – навчання та перевірки моделі за допомогою вбудованих можливостей використаних бібліотек.

На осі абсцисс зображено епохи, на осі ординат – точність для кожної епохи. Крапками зображено точність при навчанні, а лінією – точність при перевірці моделі (див. рис. 3.4).

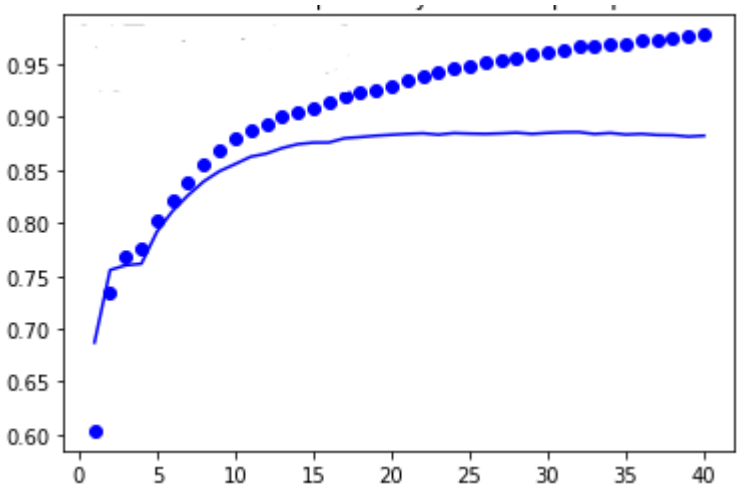


Рисунок 3.4 – Графік залежності точності від епох

Під час навчання втрати зменшуються, а точність збільшується з кожною наступною епохою. Але для перевірки моделі після двадцятої епохи точність перестає рости, що є ознакою перенавчання моделі – модель показує кращі результати на даних для навчання, аніж на нових даних для перевірки моделі. Тому має сенс зменшити кількість епох до 20.

Результати для 20 епох майже не відрізняються від результатів для 40 епох. Точність стала трохи більшою, але різниця незначна (див. рис. 3.5).

```
>>> print(results)
25000/25000 [==============================] - 1s 52us/sample - loss: 0.3365 - acc: 0.8706
[0.336505445151329, 0.8706]
```

Рисунок 3.5 – Результати для 20 епох

Отже, використання згорткової нейронної мережі для визначення тональності відгуків до фільмів показало досить прийнятний результат у 87% точності.

3.3 Дослідження ефективності застосування LTSM

Для роботи з LTSM нейронною мережею використовується не готовий набір даних, підготовлений для тренування та тестування нейронної мережі, тому до етапів дослідження додаються початкові етапи підготовки даних.

Загалом дослідження складається з наступних етапів:

– обробка даних – перетворення в нижній регістр, видалення розділових знаків;

– токенізація – створюється список слів за частотою використовування, кожне слово у відгуку замінюється на відповідне число (те є саме, що робилося для попереднього етапу дослідження);

– аналіз довжини відгуків, приведення усіх відгуків до спільної середньої довжини;

– визначення мережевої архітектури LSTM;

– побудування класу моделі;

– навчання мережі;

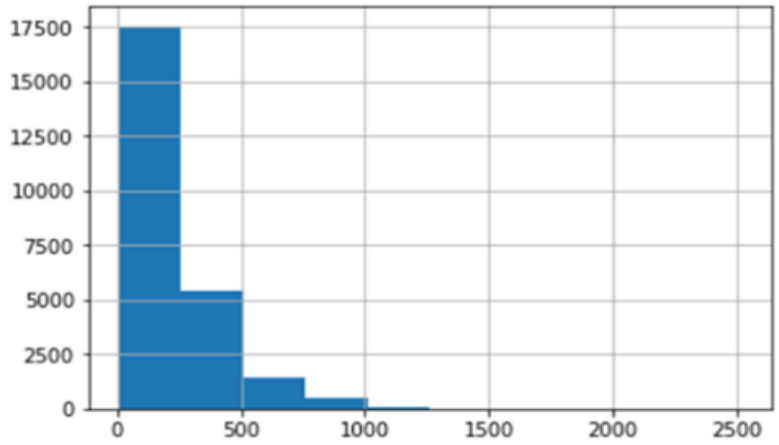
– тестування мережі.

Розглянемо детальніше етапи в-е, оскільки перші два етапи є такими ж самими, як для попереднього дослідження.

Для того, щоб визначити середню довжину відгуків, використовується бібліотека pandas. Нижче зображено приклад програмого коду для виявлення середньої довжини відгуків та наочного відображення отриманих даних у вигляді гістограми.

```
import pandas as pd
reviews_len =
    [len(x) for x in reviews_int]
pd.Series(reviews_len)
    .hist()
plt.show()
pd.Series(reviews_len)
    .describe()
```

Для більшої наочності результатів побудована гістограма (див. рис. 3.6). На осі абсцисс зображено кількість слів, а а осі ординат - кількість відгуків, що мають довжину у заданому проміжку з кроком в 250 слів.




```
count    25001.000000
mean       240.798208
std        179.020628
min          0.000000
25%        130.000000
50%        179.000000
75%        293.000000
max       2514.000000
dtype: float64
```

Рисунок 3.6 – Гістограмма середньої довжини відгуків

Отже, як можна побачити з результатів, середня довжина відгуків – 241 символ. Як і у минулому дослідженні, відгуки приводяться до загальної довжини. Ці перетворення проводяться на обох наборах даних: на тренувальному та на тестовому. В результаті ми маємо два набори відгуків однакової довжини, в яких слова закодовані цифрами. Таким чином, дані підготовлені для подальшого використання нейронною мережею.

Далі будується власне нейронна мережа та клас моделі. Спочатку, визначаються наступні гіперпараметри:

– lstm_size: Кількість одиниць у прихованих шарах у клітинах LSTM. Зазвичай більше значення краще с точки зору ефективності. Загальні значення 128, 256, 512 і т.д., але використовується середня довжина відгуку;

– lstm_layers: кількість шарів LSTM в мережі. Я б почав з 1, а потім додавав більше, якщо я недостатньо підібраний;

– batch_size: кількість відгуків для передачі мережі в одному навчальному пропуску. Як правило, це має бути встановлено настільки високим, як можна перейти без пам'яті;

– learning_rate: швидкість навчання.

Обрано наступні параметри:

```
lstm_size = 256
lstm_layers = 2
batch_size = 1000
learning_rate = 0.01
```

В загальному вигляді, архітектура нейронної мережі складається з наступних шарів:

– embedding шар, що перетворює слова (цілі числа) у вектори певного розміру;

– шар LSTM: визначається вимірами схованого стану і кількістю шарів;

– повністю підключений шар: що відображає вихід LSTM-шару на бажаний розмір виводу;

– рівень активації сигмоїдів: перетворює всі вихідні значення у значення від 0 до 1

– вихідні дані: Сигмоподібний вихід з останнього тимчасового кроку розглядається як кінцевий вихід цієї мережі.

Далі визначається клас моделі (повний код класу моделі розміщено у додатках), після чого проходить створення інстанції нейронної мережі і проходить тренування на тренувальних даних. Після закінчення тренування проводиться перевірка на тестових даних.

```
    test_acc = []
    with tf.Session() as sess:
        saver.restore(sess,
"checkpoints/sentiment_manish.ckpt")
        test_state = sess.run(cell.zero_state(batch_size,
tf.float32))
        for ii, (x, y) in enumerate(get_batches(test_x,
test_y, batch_size), 1):
            feed = {inputs_: x,
                    labels_: y[:, None],
                    keep_prob: 1,
                    initial_state: test_state}
            batch_acc, test_state =
    sess.run([accuracy, final_state],
    feed_dict=feed)
            test_acc.append(batch_acc)
```

```
        print("Test accuracy:
{:.3f}".format(np.mean(test_acc)))
```

Після декількох випробувань було виявлено, що десь після 20 епохи проходить перенавчання нейронної мережі і точність передбачень перестає зростати, тому було прийнято рішення зупинитися на 15 епохах.

Результати тестування зображено на рисунку 3.7.

```
>>> print("Test accuracy: {:.3f}".format(test_acc))
Test accuracy: 0.884
```

Рисунок 3.7 - Результати тестування для LSTM-мережі

Як можна побачити з результатів, застосування LSTM-мережі дало результат трохи кращий, ніж застосування простої згорткової мережі.

3.4 Дослідження використання попередньо навченої моделі

У якості попередньо навченої моделі буде використовуватися Google’s BERT. Оскільки модель вже навчена, в даному випадку нам не треба власноруч проводити тренування нейронної мережі, і модель одразу можна використовувати для виявлення тональності текстів.

Для того, щоб використовувати попередньо навчену модель, потрібно лише завантажити її з офіційного гітхаб-репозиторія.

Модель потребує набір з двох колонок, одна з яких – власне відгук, а друга – його полярність. Також треба задати вхідні параметри:

```
myparam =
    {
          "DATA_COLUMN": "text",
          "LABEL_COLUMN": "sentiment",
          "LEARNING_RATE": 2e-5,
          "NUM_TRAIN_EPOCHS":20
     }
```

Перші два параметри – це назви колонок у наборі даних для тестування. LEARNING_RATE залишимо за замовчуванням, а кількість епох вкажемо таку ж саму, як для попередніх досліджень.

Як і для минулих досліджень, треба провести токенізацію даних: кожне слово в відгуці представити у вигляді числа, після чого представити відгуки у вигляді векторів, так само, я у минулих дослідженнях. Для токенізації даних було використано такі ж самі методи, як для дослідження використання LSTM нейронної мережі.

Лейбли, що вказують на полярність тексту (позитивна чи негативна) знову мають бути представлені у вигляді бінарного флагу 0 чи 1.

Для того, щоб отримати результат, треба лише викликати функцію run_on_dfs, куди передається набір даних та визначені параметри. Результати виконання визначення полярності тексту зображено на рисунку 3.8.

```
{'auc': 0.856,
 'eval_accuracy': 0.856,
 'f1_score': 0.852459,
 'false_negatives': 84.0,
 'false_positives': 60.0,
 'global_step': 187,
 'loss': 0.530802,
 'precision': 0.8739496,
 'recall': 0.832,
 'true_negatives': 440.0,
 'true_positives': 416.0}
```

Рисунок 3.8 - Результати аналізу полярності відгуків з BERT

Як можна побачити з результатів, точність визначення полярності трохи нижча, ніж для попередніх методів. Це пояснюється тим, що використану модель було попередньо навчено на іншому датасеті, і нейронною мережею не було виявлено ознаки, специфічні для власне відгуків на фільми.

3.5 Додаток для виявлення тональності заданого користувачем тексту

Для бінарної класифікації тональності тексту, довільно заданого користувачем, було використано три попередньо треновані нейронні мережі: згорткова, LSTM-мережа та модель Google’s BERT.

Додаток реалізовано у вигляді веб-сторінки з полем для вводу текста та dropdown-елементом для вибору конкретної моделі (якщо жодна конкретна модель не була вибрана, то користувачеві видається результат для усіх трьох варіантів (див. рис. 3.9).

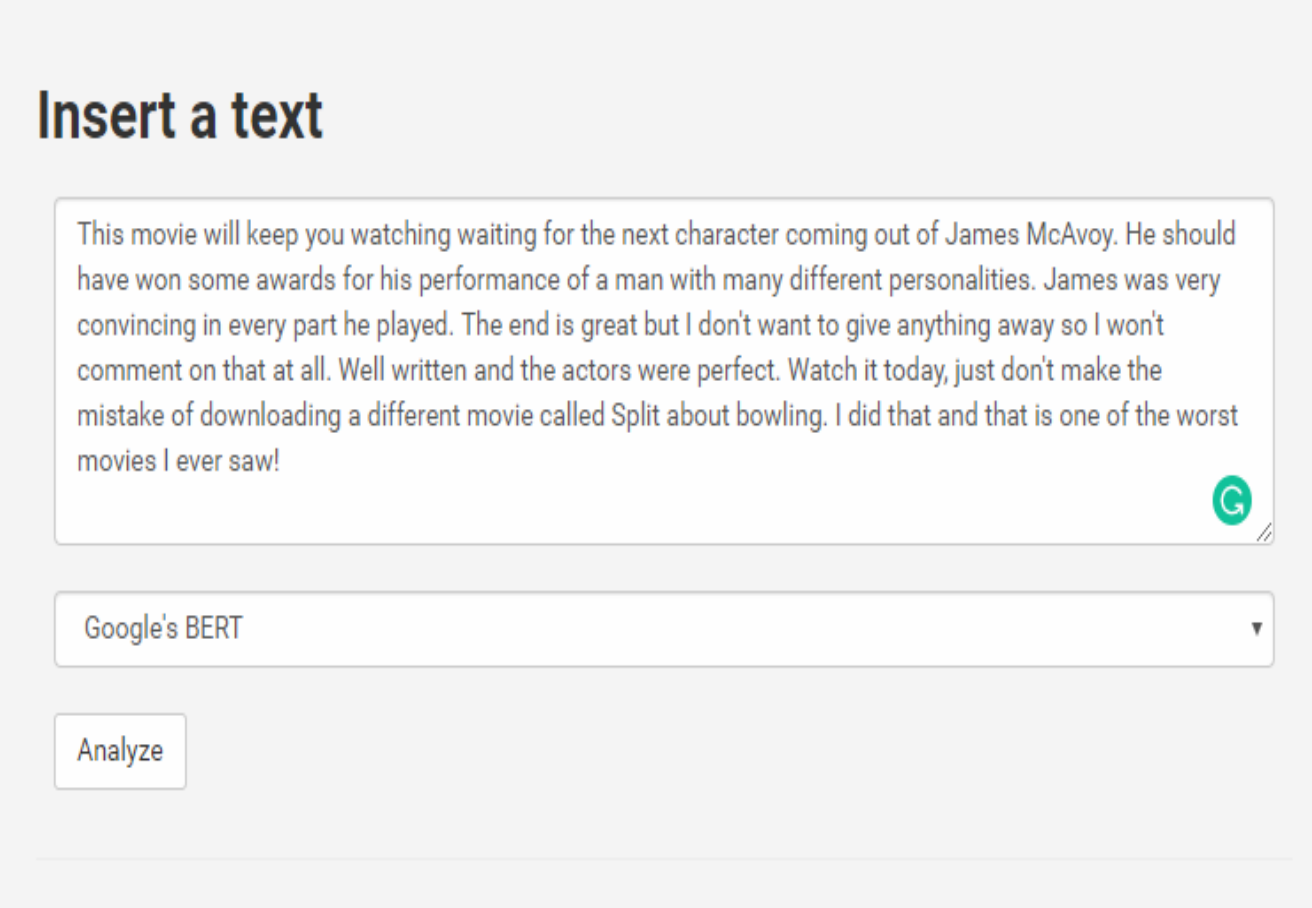


Рисунок 3.9 - Веб-інтерфейс для виявлення полярності тексту

Користувацький інтерфейс побудовано з використанням jacascript-бібліотеки ReactJS, що дозволяє швидко збудувати клієнтський додаток і дає можливість відправляти запити на сервер.

Для проведення дослідження було обрано не лише відгуки на фільми з бази IMDB, а також випадкові тексти з наступних датасетів:

– Twitter US Airline Sentiment – цей набір даних містить дані Twitter про авіакомпанії США, які були зібрані з лютого 2015 року. Автори класифікували твіти як позитивні, негативні та нейтральні твіти (будуть використані лише позитивні та негативні);

– Paper Reviews – у цьому наборі даних містяться висловлені у наукових публікаціях речення, які позначають позитивні відгуки від міжнародної конференції з обчислювальної техніки та інформатики;

– Amazon Reviews for Sentiment Analysis – цей набір даних складається з декількох мільйонів відгуків клієнтів Amazon (вхідний текст) і рейтингів зірок (вихідні позначки) для навчання способу підготовки швидкого тексту для аналізу настроїв. Оскільки додаток працює з бінарною класифікацією, відгуки с рейтингом 4-5 вважаються позитивними, а з рейтингом 1-3 – негативними. Датасет не містить нейтральних відгуків.

Результати досліджень приведено у таблиці 3.1.

Таблиця 3.1 - Результати досліджень для різних датасетів

Датасет	Згорткова мережа	LSTM-мережа	BERT
Відгуки IMDB	0,86	0,87	0,85
Twitter US Airline Sentiment	0,67	0,701	0,83
Paper Reviews	0,76	0,79	0,84
Amazon Reviews	0,78	0,81	0,84

Як можна побачити з результатів дослідження, власні моделі краще спрацювали на тому ж датасеті, на якому проводилося тренування, в той час як для текстів іншої тематики власні треновані моделі показали гірший результат, що

можна пояснити тим, що власні треновані моделі для тренування використовували менший корпус текстів, до того ж з однієї предметної області. Найгірший результат ці мережі показали у випадку з даними з твітера. Це можна пояснити тим, максимальна довжина повідомлення в твіттері - 280 знаків, в той час як середня довжина відгуку на IMDB - 240 слів.

Найкраще з новими датасетами впоралатся нейронна мережа BERT. Це можна пояснити тим, що вона натренована на менш специфічному датасеті та на більшому корпусі статей.

Однак при розширенні тренувального датасету для LSTM-мережі, потенційно можна досягти тих самих, або й навіть кращих результатів. Доцільним є використання корпусу даних різної довжини, а також з різних предметних областей.

ВИСНОВКИ

Аналіз настроїв призначений для автоматизованого виявлення в текстах емоційно забарвленої лексики і емоційної оцінки авторів (думок) по відношенню до об'єктів, мова про які йде в тексті. Це контекстний видобуток тексту, який ідентифікує та видобуває суб'єктивну інформацію у вихідному матеріалі.

Аналіз тональності текстів здатний допомогти розібратися в законах, за якими живе природна мова і навчити комп'ютер сприймати його на рівні, наближеному до людського. Також аналіз тональності здатний значно покращити якість перекладів.

Аналіз тональності текстів допомагає бізнесу зрозуміти соціальні настрої свого бренду, продукту або послуги, контролюючи онлайн-розмови. Він може бути використаний для проведення досліджень ринку, аналітики продуктів. Інша сфера, де аналіз тональності може бути влучно використаний - підтримка користувачів та аналіз зворотнього зв’язку, а також моніторинг соціальних медіа.

Нейронні мережі є потужним інструментом для розв’язання задачі аналізу тональностей тексту. Якщо необхідно проводити аналіз текстів конкретної предметної області (наприклад, аналіз звернень до служби підтримки), то більш ефективним є використання нейронної мережі, натренованої на тренувальних даних тієї ж самої предметної області.

У випадку ж, коли система повинна визначати тональність довільних текстів, для яких не можна виділити спільне джерело чи тематику, має сенс використовувати підхід Transfer Learning, тобто використовувати попередньо тренеровані моделі. У результаті дослідження було виявлено, що чим більш варіативними є дані для навчання нейронної мережі, тим більший відсоток її ефективності для довільних текстів.

У практичних задачах можуть використовуватися як теоретичні, так і практичні результати досліження, оскільки частими є задачі як визначення тональності текстів спільного напрямку, так і цілком довільних. Отримані

натреновані моделі можуть використовуватися одразу для розв’язання задач аналізу тональностей тексту на практиці.

У подальшому є сенс розвивати дослідження в двох напрямах. Перший – розширення оцінки тональності від бінарної до множинної, а також виділення емоціонально нейтральних текстів. Більша кількість варіантів тональності дозволить більш точно розв’язувати задачі, пов’язані, наприклад, з визначенням приоритету повідомлень в службу підтримки, і покращувати клієнтский сервіс.

Інший потенційно цікавий напрям продовження дослідження – використання Transfer learning і попередньо навчених моделей; метою для цього напряму є побудування універсальної моделі для аналізу тональностей, яка буде показувати однаково високі результати для текстів, різних за змістом, лексиконом, довжиною та ін.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bing Liu, Sentiment Analysis: Mining Opinions, Sentiments, and Emotions – Cambridge University Press; 1 edition – 2015, 383 с.

2. Cambria, E., Das, D., Bandyopadhyay, A Practical Guide to Sentiment Analysis, Springer – 2017, 199 с.

3. The best sentiment analysis tools / TalkWalker, URL: https://www.talkwalker.com/blog/best-sentiment-analysis-tools (Дата звернення: Квітень 12, 2019)

4. Gerardus Blokdyk, Sentiment Analysis a Complete Guide, 5starcooks – 2018, 126 с.

5. Emotion and Sentiment Analysis: A Practitioner's Guide to NLP, URL: https://www.kdnuggets.com/2018/08/emotion-sentiment-analysis-practitioners-guide-nlp-5.html (Дата звернення: Березень 30, 2019)

6. Passenger dragged off overbooked United flight, URL: https://edition.cnn.com/2017/04/10/travel/passenger-removed-united-flight-trnd/index.html Дата звернення: Березень 23, 2019)

7. Soudamini Hota, Sudhir Pathak, KNN classifier based approach for multi-class sentiment analysis of twitter data, Independently publisher – 2017, 124 с.

8. Sentiment Analysis: learn everything you need to know, URL: https://monkeylearn.com/sentiment-analysis/ (Дата звернення: Березень 23, 2019)

9. Trump vs Hillary: Sentiment analysis on Twitter mentions, URL: https://monkeylearn.com/blog/trump-vs-hillary-sentiment-analysis-twitter-mentions/ (Дата звернення: Березень 30, 2019)

10. IMDB Movies Review dataset, URL: https://www.kaggle.com/iarunava/imdb-movie-reviews-datasetb (Дата звернення: Квітень 5, 2019)

11. Francois Chollet, Deep Learning with Python Languagr, 1st Edition – 2017, 384

12. Pradeep Pujari, Md. Rezaul Karim, Practical Convolutional Neural Networks, Packt Publishing – 2018, 218 с.

13. Frank Millstein, Deep Learning: 2 Manuscripts - Deep Learning With Keras And Convolutional Neural Networks In Python Paperback – 2018, 260 с.

14. Bianchi, F.M., Maiorino, E., Recurrent Neural Networks for Short-Term Load Forecasting, SpringerBriefs – 2017, 72 с.

15. Frank Millstein, Python Machine Learning: Introduction To Machine Learning With Python, Kindle Edition – 2018, 134 с.

16. Simeon Kostadinov. Recurrent Neural Networks with Python Quick Start Guide: Sequential learning and language modeling with TensorFlow. Paperback – 2018, 122 с.

17. Jonathon Chambers, Recurrent Neural Networks for Prediction: Learning Algorithms,Architectures and Stability, Leicester – 2011, 105 с.

18. LazyProgrammer, Deep Learning: Recurrent Neural Networks in Python: LSTM, GRU, and more RNN machine learning architectures in Python and Theano (Machine Learning in Python), Kindle Edition – 2016, 56 с.

19. Ian Goodfellow, Yoshua Bengio, Deep Learning (Adaptive Computation and Machine Learning series), The MIT Press – 2016, 775 с.

20. Transfer Learning / University of WISCONSIN. URL: http://pages.cs.wisc.edu/~shavlik/abstracts/torrey.handbook09.abstract.html (Дата звернення: Квітень 25, 2019)

21. BERT Explained: State of the art language model for NLP / Towards Data science, URL: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270 (Дата звернення: Травень 12, 2019)