



Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Комп'ютерні науки  
(повна назва)

Кафедра \_\_\_\_\_ Програмної інженерії  
(повна назва)

Рівень вищої освіти \_\_\_\_\_ другий (магістерський)

Спеціальність \_\_\_\_\_ 121 – Інженерія програмного забезпечення  
(код і повна назва спеціальності)

Тип програми \_\_\_\_\_ освітньо-наукова програма  
(освітньо-професійна або освітньо-наукова)

Освітня програма \_\_\_\_\_ Інженерія програмного забезпечення  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав.кафедри \_\_\_\_\_  
(підпис)

« 26 » \_\_\_\_\_ березня 2021 р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента \_\_\_\_\_ Назаренка Дмитра Сергійовича  
(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження методів нейронних мереж та розробка гібридного підходу для розпізнавання емоцій за текстом»

затверджена наказом університету від 26.03.2021 № 385 Ст

2. Термін подання студентом роботи до екзаменаційної комісії «17» травня 2021 р.

3. Вихідні дані до роботи методи та порівняння нейронних мереж, розробка гібридного підходу для розпізнавання емоцій за текстом

4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз предметної галузі і постановка задачі, огляд методів глибоке навчання, методи нейронних мереж для розпізнавання емоцій за текстом, розробка та порівняння архітектур нейронних мереж, розробка гібридного підходу

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, ілюстрацій (слайдів) мета роботи, постановка задачі, методи нейронних мереж, розроблені архітектури, опис отриманих результатів, гібридний підхід, демонстраційні матеріали

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз предметної галузі	26.03 – 02.04.2021	<i>Виконано</i>
2	Аналіз методів розпізнавання емоцій	03.04 – 10.04.2021	<i>Виконано</i>
3	Розробка методів та виконання експериментальної частини	11.04 – 18.04.2021	<i>Виконано</i>
4	Підготовка пояснювальної записки	19.04 – 04.05.2021	<i>Виконано</i>
5	Спецчастина	19.04 – 11.05.2021	<i>Виконано</i>
6	Підготовка презентації та доповіді	27.04 – 11.05.2021	<i>Виконано</i>
7	Нормоконтроль, рецензування	12.05 – 15.05.2021	<i>Виконано</i>
8	Занесення матеріалів в електронний архів	15.05.2021	<i>Виконано</i>
9	Попередній захист	15.05.2021	<i>Виконано</i>
10	Допуск до захисту у зав. кафедри	17.05.2021	<i>Виконано</i>

Дата видачі завдання 26 03 2021р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Афанасьєва І.В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 85 с., 6 рис., 23 табл., 29 джерел, 5 додатків.

ГЛИБОКЕ НАВЧАННЯ, ДАНІ, ЕМОЦІЇ, КЛАСИФІКАЦІЯ, НЕЙРОННІ МЕРЕЖІ, ТЕКСТ, KERAS

Метою роботи є методи та підходи в нейронних мережах, проектування, розробка та порівняння роботи методів глибокого навчання в сфері розпізнавання емоцій за текстом.

У якості методів розробки та аналізу була використана мова програмування Python, фреймворк для розробки нейронних мереж Keras, середовище розробки Jupyter Notebook та операційна система Ubuntu 18.10.

Робота присвячена вивченню найпопулярніших методів глибокого навчання – обробці природної мови, зокрема, вилученню емоцій з тексту. Було порівняно кілька типів ембедінгів та різні архітектури нейронних мереж для вирішення проблеми класифікації емоцій із тексту.

У роботі було розроблено гібридний підхід, який базується на суміщенні робіт двох незалежних нейронних мереж, та проаналізовано його роботу на тестових наборах даних.

Було проведено порівняння між звичайними та гібридним підходами. В результаті було знайдено підхід для створення нейронних мереж, який забезпечує кращі результати на текстових даних.

CLASSIFICATION, DATA, DEEP LEARNING, EMOTIONS, NEURAL NETWORKS, SENTIMENTS, TEXT, KERAS

The aim of the work is algorithms and approaches in neural networks, design, development and comparison of deep learning methods in the field of emotion recognition by text.

Python programming language, Keras neural network development framework, Jupyter Notebook development environment and Ubuntu 18.10 operating system were used as development and analysis methods.

The work is devoted to the study of the most popular methods of in-depth learning – the processing of natural language, in particular, the extraction of emotions from the text. Several types of word embedding and different neural network architectures were compared to solve the problem of classifying emotions from text.

The paper was developed hybrid approach based on combining two independent robot neural networks, and analyzes its performance on the test data sets.

A comparison was made between conventional and hybrid approaches. As a result, an approach was found to create neural networks that provides better results on textual data.

Я, Назаренко Дмитро Сергійович, студент гр. ПЗМ-19-1, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів нейронних мереж та розробка гібридного підходу для розпізнавання емоцій за текстом», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIAr KhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений (а) з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

## ЗМІСТ

Зміст .....	6
Скорочення та умовні позначки .....	8
Вступ.....	9
1 Аналіз предметної галузі .....	11
1.1 Аналіз літератури .....	11
1.2 Методи нейронних мереж.....	14
1.3 Постановка задачі.....	20
2 Опис методів досліджень.....	20
2.1 Набори даних.....	21
2.2 Ембедінги слів .....	24
2.3 Метрики .....	26
2.4 Моделі.....	26
3 Експериментальні дослідження.....	29
3.1 Архітектура BiLSTM.....	29
3.2 Архітектура CNN .....	33
3.3 Архітектура BiGRU.....	37
3.4 Гібридний підхід .....	41
3.5 Комплексна оцінка ефективності моделей .....	44
Висновки.....	46
Перелік джерел посилання .....	47
Додаток А Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії .....	50

Додаток Б Звіт результатів перевірки кваліфікаційної роботи на унікальність тексту .....	51
Додаток В Слайди презентації .....	52
Додаток Г Апробація результатів роботи .....	62
Додаток Д Фрагмент коду програми.....	71
Додаток Е Експертний висновок нормоконтроль .....	84

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

BiGRU – Bidirectional Gated Recurrent Unit

BiLSTM – Bidirectional Long Short-Term Memory

CBOW – Continuous Bag of Words

CNN – Convolutional neural network

DNN – Dense Neural Network

EWE – Emotional Word Embedding

GloVe – Global Vectors for Word Representation

GRU – Gated Recurrent Unit

LSTM – Long Short-Term Memory

RNN – Recurrent neural network

## ВСТУП

Виявлення емоцій і розпізнавання тексту в сучасному світі це поле мейнстріму. За допомогою мовної обробки (NLP) можна зробити цінний внесок у дослідження. Сьогодні записи / дані приймають різні форми публікацій у соціальних мережах, новинних статей, оглядів користувачів тощо, а вміст коротких текстів може бути корисним ресурсом для видобутку тексту для розкриття різноманітних питань, які можна вирішити з текстом, зокрема емоції.

Відсутність розумних знань ускладнює для комп'ютера розпізнавати та генерувати емоції. Тому було проведено значне дослідження щодо розпізнавання емоцій. Розпізнавання емоцій поділяється на три основні категорії: розпізнавання за мімікою, у розмові та в тексті [1]. Розпізнавання емоцій у тексті означає завдання автоматичного присвоєння емоції тексту, вибраному із набору заздалегідь визначених міток емоцій. Цей напрям дуже важливий, тому що текст є основним засобом людської взаємодії з комп'ютером.

Виявлення та розпізнавання емоцій з тексту – недавня область дослідження, яка тісно пов'язана з аналізом настроїв. Аналіз настрою спрямований на виявлення позитивних, нейтральних або негативних почуттів з тексту, тоді як аналіз емоцій спрямований на виявлення та розпізнавання типів почуттів за допомогою вираження текстів, таких як гнів, щастя, радість, любов, смуток, здивування, страх, провина, сором, ненависть, і огиди.

У випадку аналізу настроїв, завдання розпізнавання можна вирішити, використовуючи лексиконні методи, машинне навчання або підхід на рівні концепції. В роботі досліджується, як можна досягти цього завдання за допомогою підходу машинного навчання, використовуючи прийоми глибокого навчання.

Сьогодні в Інтернеті є величезна кількість текстових даних. Процес обробки емоцій в текстових даних користується багатим попитом для різних цілей, особливо в таких сферах як бізнес. Застосування розпізнавання емоцій у тексті

можна знайти в бізнесі, психології, освіті та багатьох інших сферах, де існує потреба зрозуміти та інтерпретувати емоції.

Існують основні теорії емоцій, які були розроблені щодо того, як деякі емоції впливають на сприйняття. Є багато робіт, які досягли достатніх результатів у галузі розпізнавання емоцій у тексті. Покращення результату в розпізнаванні емоцій за допомогою реальних даних все ще залишається величезною проблемою з кількох причин.

Більшість методів машинного навчання надмірно покладаються на особливості ручної роботи, які вимагають великої кількості ручного проектування та налаштування, а це трудомістко та витратно. Однак цій проблемі дуже допомагає ідея про використання глибокого навчання.

Останнім часом створюється все більше архітектур нейронних мереж, які можуть використовувати як семантичну / синтаксичну, так і емоційну інформацію, приймаючи заздалегідь навчені ембедінги слів для досягнення більш ефективних результатів.

У роботі проведено порівняльний аналіз найсучасніших архітектур нейронних мереж, в результаті якого визначено ефективність роботи з текстовими даними для вилучення емоційної складової. Також було розроблено гібридний підхід та побудовано моделі на його основі.

Було проведено комплексну оцінку ефективності моделей із використанням реальних наборів даних. Було проведено порівняльний аналіз звичайних моделей та моделей побудованих за гібридним принципом.

Експерименти виявили архітектури, які демонструють найкращу здатність розпізнавати емоції в тексті.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

### 1.1 Аналіз літератури

Глибоке навчання – це галузь машинного навчання, в якій програми вчаться на досвіді та розуміють світ з точки зору ієрархії понять, де кожна концепція визначається з точки зору її відношення до більш простих концепцій. Цей підхід дозволяє програмі вивчати складні концепції, будуючи їх на основі більш простих [1]. Найбільш часто використовуваною моделлю глибокого навчання є модель з довгою короткочасною пам'яттю (LSTM). Модель з довгою короткочасною пам'яттю – це форма рекурентної нейронної мережі (RNN), яка може обробляти довгострокові залежності. LSTM вирішує проблему зникнення або розриву градієнтів, які завдають шкоди RNN.

Було розглянуто кілька спроб застосувати глибоке навчання до розпізнавання емоцій. Було досліджено деякі впроваджені підходи глибокого навчання, щоб вирішити, які підходи є найсучаснішими.

Надійне представлення твіту було запропоновано Мейшері та Дей [3]. Використовуючи різні попередньо навчені ембедінги, було побудовано дві паралельні архітектури для створення представлення. Матриця ембедінгу була сформована за допомогою emoji2vec [4], GloVe [5] та вбудовувань на рівні символів у першій архітектурі. Для подання отриманої матриці використовували BiLSTM [6]. Кожен раз вихідний результат кроку подавався на рівень уваги (attention layer) [7]. Матриця ембедінгу була розроблена другою архітектурою, яка використовувала попередньо навчені ембедінги GloVe, навчені на даних з Twitter. Ця матриця була подана в іншій BiLSTM. Виходи двох архітектур були об'єднані та подані у дві повністю з'єднані мережі. У змаганнях SemEval-2018 їх модель вийшла на друге місце серед команд.

Щоб вирішити розпізнавання емоцій із декількома мітками, Wang et al. [8] використовував згорткову нейронну мережу (CNN). У тестах використовувались набір даних китайського блогу Ren CECps [9] та завдання NLPCC 2014 Emotion

Analysis у китайському тексті Weibo [10]. Експериментальні результати показали, що CNN перевершує базові показники та досягає чудової ефективності за допомогою ембеддінгів слів.

Розпізнавання емоцій у тексті було сформульовано як проблему бінарної класифікації Арміном Сейедітабарі [11]. В якості моделей ембеддінгів слів використовувались ConceptNet Numberbatch [12] та FastText [13]. Шар BiGRU отримує шар ембеддінгу. Після цього виконується операція пулінгу над даними та результати подаються в DNN, і класифікація здійснюється сигмоподібним шаром. Отримані дані показують, що моделі глибокого навчання можуть засвоїти більш глибокі особливості, що призведе до значного поліпшення ефективності.

Модель глибокого навчання для виявлення багатозначних емоцій у мікроблогах була запропонована Pathum Rathnayaka [14]. На першому етапі текстові дані були попередньо оброблені. Після цього вхідні дані були подані до вхідного шару, який являв собою попередньо навчений ембеддінг слів GloVe. Результати подаються на два шари BiGRU. Потім вбудований шар і вихідний сигнал першого шару BiGRU подаються в перший шар. Шар ембеддінгу, вихід першого і другого шару BiGRU також подаються у другий шар. Для проведення класифікації два шари об'єднуються і подаються в DNN з функцією активації сигмовидної форми.

Останнім часом пропонуються інші комбіновані архітектури для задач розпізнавання емоцій на основі тексту. Наприклад, комбінований підхід рекуррентних згорткових нейронних мереж (RCNN) досягає конкурентних результатів за допомогою відрегульованих ембеддінгів GloVe та емоційних ембеддінгів. Експериментальні результати показують, що доопрацьовані ембеддінги GloVe працюють помітно краще, ніж контекстуальні ембеддінги, оскільки завдання розпізнавання емоцій сильно залежить від методу обробки емоцій та розміру словника слів.

Двонаправлений LSTM, також відомий як BiLSTM, – це модель обробки послідовностей, яка складається з двох LSTM, одна з яких приймає вхід в одному напрямку, а інша – в іншому. BiLSTM суттєво збільшують обсяг даних, доступних

для мережі, надаючи алгоритму більше значення (наприклад, знаючи, які слова безпосередньо слідують і передують слову в реченні).

CNN (Convolutional Neural Network) були вперше розроблені в галузі обробки зображень нейронної мережі, де вони досягли новаторських результатів у розпізнаванні об'єктів із заздалегідь визначеної категорії (наприклад, кішка, велосипед тощо). Згортання та об'єднання – це дві операції, які можна розглядати як екстрактори функцій у згортковій нейронній мережі. Для класифікації текстових даних використовується одновимірний масив, що представляє текст у випадку завдань NLP, тобто коли застосовується до тексту, а не до зображень. У цьому випадку архітектура згорткової нейронної мережі модифікована до одновимірних операцій згортання та об'єднання.

Рекуррентна нейронна мережа із вентиляним рекуррентним вузлом, або GRU, є різновидом рекуррентної нейронної мережі. Це схоже на LSTM, за винятком того, що він має лише два вузла: вузел сбросу та вузел оновлення, і він не має вихідного вузла. GRU простіші та швидші в навчанні, ніж LSTM, оскільки вони мають менше параметрів. Двонаправлений GRU, також відомий як BiGRU, – це модель обробки послідовностей, що складається з двох GRU. Один зчитує дані у прямому напрямку, а інший у зворотньому.

Як було згадано вище, методи, засновані на глибокому навчанні, переважно використовують вектори слів. Вбудовані слова – це тип подання слів, який пов'язує розуміння машиною мови з людською. Вони вивчили текстові подання у  $n$ -мірному просторі, де слова з подібними значеннями представлені однаково. Іншими словами, два споріднені слова представлені майже однаковими векторами, які розташовані дуже близько один до одного у векторному просторі. Word2Vec [15], GloVe [16] та FastText [17] є найбільш широко використовуваними методами ембеддінгів. Word2Vec – одна з перших обчислювальних потужних моделей для вивчення подань трильйонів слів. Це значно перевершило різні  $n$ -грамові моделі [18]. Пізніше був опублікований ембеддінг GloVe, що означає глобальні вектори для подання слів. Він показав кращий перформанс, ніж ембеддінг Word2Vec, тому що він дізнавався загальний контекст з тексту, а не різні локальні контексти, які

вивчав Word2Vec. FastText нещодавно був розроблений для виявлення та вивчення подання слів. Слова розглядаються як найменші елементарні одиниці в Word2Vec та GloVe. FastText застосовує інший підхід, розглядаючи кожне слово як набір n-грамових символів, і воно є більш універсальним, ніж інші дві моделі, оскільки може обробляти незвичайні терміни, яких немає у словниках. Аналіз сентиментного ембедінгу (EWE) [19] базується на значно меншому корпусі текстових даних. Цей ембедінг можна легко застосувати до завдань, пов'язаних з емоціями, які в значній мірі можуть подолати обмеження словникового запасу емоцій. Однак воно обмежується семантичними та синтаксичними зв'язками.

На основі аналізу було вирішено використовувати 3 основні методи в побудові нейронних мереж: BiGRU, BiLSTM та CNN. Для роботи з текстовими даними було вирішено використовувати ембедінги в аналізованих моделях: ембедінг Word2Vec, GloVe, FastText та EWE.

Було прийнято рішення розробити гібридний підхід, використовуючи поєднання архітектур, які будуть відповідати за різні складові текстових даних. Для порівняльного аналізу потрібно збудувати, натренувати та протестувати моделі на основі цих архітектур.

## 1.2 Методи нейронних мереж

Ембедінг – це відносно низьковимірний простір, в який можна перекласти вектори високих розмірів. Ембедінги полегшують навчання на великих вхідних даних, таких як розріджені вектори, що представляють слова. В ідеалі, ембедінг зберігає частину семантики вхідних даних, розміщуючи семантично подібні дані близько в просторі ембедінгу. Ембедінг можна вивчити та повторно використовувати в різних моделях.

Загалом, ембедінг є формою подання слів [14]. Це допомагає зв'язати людину і машину в розумінні мови. Цей тип шарів забезпечує розподілене

представлення тексту в  $n$ -мірному просторі. Ембеддінг слів – це сімейство методів обробки природної мови, спрямованих на відображення семантичного значення в геометричному просторі. Це реалізується шляхом асоціації числового вектора з кожним словом із словника так, що відстань між будь-якими двома векторами охоплює частину семантичних відношень між двома спорідненими словами. Наприклад, «кокос» і «білий ведмідь» – це слова, які семантично досить різні, і ембеддінг представятиме їх як вектори, які будуть розташовані на великій відстані один від одного; але слова «кухня» та «вечеря» є спорідненими словами, тому вони будуть розташовані ближче одне до одного.

У роботі розглядаються 4 основних типу векторного представлення слів: Word2Vec, GloVe, FastText та EWE.

Word2Vec – це метод для ефективного створення вбудованих слів за допомогою двошарової нейронної мережі. Його розробив Томаш Міколов в Google у 2013 році як відповідь на підвищення ефективності навчання ембеддінгу на основі нейронних мереж і з тих пір це стало фактичним стандартом для розробки попередньо навченого ембеддінгу слів [15].

Вхідні дані Word2vec – це текстовий корпус, а його вихідні дані – це набір векторів, відомих як вектори об'єктів, що представляють слова в цьому корпусі. Хоча Word2vec не є глибокою нейронною мережею, він перетворює текст у числову форму, яку глибокі нейронні мережі можуть зрозуміти.

Цільова функція Word2vec змушує слова, що мають подібний контекст, мати подібні ембеддінги. Отже, у цьому векторному просторі ці слова справді близькі. Математично, косинус кута між такими векторами повинен бути близьким до 1, тобто кутом, близьким до 0.

Word2vec – це не єдиний алгоритм, а поєднання двох методів – CBOW (Continuous Bag of Words) та skip-gram model. Обидва вони є неглибокими нейронними мережами, які відображають слова до цільової змінної, яка також є словом. Обидва ці прийоми вивчають ваги, які виступають як подання векторів слів.

GloVe (глобальні вектори для представлення слів) – це альтернативний метод створення вбудованих слів. Він базується на методах розкладання матриць на матрицю слова-контексту. Основою алгоритму є побудова великої матриці інформації про спільну появу, і підрахунок кожного слова (рядка), і те, як часто це слово зустрічається в якомусь контексті (стовпці) у великому корпусі [16].

Розглянемо алгоритм GloVe:

- збираємо статистичні дані про спільне використання слів у формі матриці спільного використання слів  $X$ . Кожен елемент  $X_{ij}$  такої матриці відображає, як часто слово  $i$  з'являється в контексті слова  $j$ . Зазвичай сканується корпус наступним чином: для кожного терміна шукається терміни контексту в межах певної області, визначеної розміром вікна перед терміном та розміром вікна після терміна. Крім того, надається менше ваги для більш віддалених слів;

- визначаємо  $m$ 'які обмеження для кожної пари слів;

- визначаємо функцію витрат.

П. Бояновський запропонував новий метод ембедінгу під назвою FastText. Його ключовою зміною було використання внутрішньої структури слова для поліпшення векторних подань, отриманих методом skip-gram [17].

Для отримання векторного представлення слів одночасно використовуються моделі skip-gram та CBOW. Skip-gram ігнорує структуру слова, але в деяких мовах є складові слова, як, наприклад, в німецькому. Тому до основної моделі була додана subword-модель. Subword-модель – це уявлення слова через ланцюжки символів ( $n$ -грами) з  $n$  від 3 до 6 символів від початку до кінця слова плюс саме слово цілком. Наприклад слово замок з  $n$  дорівняно 3 буде представлено  $n$ -грамами за, зам, амо, мок, ок і послідовністю «замок». Таким чином, для моделі є різниця між послідовністю «зам» в слові замовник – і  $n$ -грамою зам зі слова замок. Такий підхід дозволяє працювати і з тими словами, які модель раніше не зустрічала.

Ознаки, отримані за допомогою розбиття на  $n$ -грами, мають величезну розмірність (тобто для тексту виходить величезна важка таблиця). Це може сповільнити роботу моделі, яка навчена на цих ознаках. Для фіксування розмірності ознак застосовується хешування ознак (спеціальна процедура, яка

дозволяє кодувати об'єкти різних розмірів за допомогою символічних ланцюжків однакової довжини). Ознаки отримують хеш-індекси, що допомагає зчитувати їх швидше.

В основі класифікатора лежить модель лінійної класифікації, яка за архітектурою схожа з моделлю SBOW. Чим більше кількість класів, тим більше час роботи лінійної моделі. Для оптимізації класифікатора використовується ієрархічний softmax заснований на алгоритмі кодування Хаффмана.

Дерево класів складається з гілок, де зверху клас, який зустрічається найбільше, а його діти – класи, які зустрічаються менше. Наприклад, у класу «Біологія» будуть діти класів «Тварини» і «Рослини» і т.д. Ймовірності класів визначаються для кожного вузла від першого вузла-батька до останнього вузла-дитини. Таким чином, у дочірнього вузла ймовірність завжди менше, ніж у його вузла-батька.

Emotion Word Embeddings (EWE) – ембеддінг слів, що проектує емоційно подібні слова в сусідні простори. Для створення цього типу ембеддінгу застосовуються архітектури рекуррентних нейронних мереж для вивчення уявлень, що усвідомлюють емоції, використовуючи великі навчальні дані. Зокрема, використовуються двонаправлені мережі довготривалої пам'яті (BiLSTM), щоб захопити семантичну інформацію між словами текстового документа, а також інформацію про емоції, надану у вигляді цільової мітки, отриманої за допомогою розмічування [19].

Слова текстового документа спочатку перетворюються у векторні подання, які потім послідовно подаються у модель BiLSTM. Потім, шляхом зворотного розповсюдження, оригінальні вектори слів оновлюються під час навчання, виробляючи збагачені емоціями ембеддінги. Архітектура моделі фіксує контекстну інформацію шляхом моделювання віддалених залежностей між словами текстового документа та інформації про емоції, що надається через цільову мітку для відображення кожного слова в просторі ембеддінгу. По-перше, створюється корпус документів, позначених емоціями, за допомогою процесу віддаленого нагляду. Потім цей корпус використовується як навчальні дані для вивчення збагачених

емоціями подань слів за допомогою BiLSTM. Іншими словами, хоча для створення навчального набору використовується маркування на рівні документа (цілі приклади), ембедінги оновлюються на рівні окремих слів.

Як зазначалося раніше, RNN мають труднощі у вивченні довгострокових залежностей. Моделі, засновані на LSTM, є розширенням для RNN, які здатні вирішити проблему зникаючого градієнта. Моделі LSTM по суті розширюють пам'ять RNN, щоб дозволити їм зберігати та вивчати довгострокові залежності входів. Це розширення пам'яті має здатність запам'ятовувати інформацію протягом більш тривалого періоду часу, що дозволяє читати, писати та видаляти інформацію зі своїх спогадів.

Модель LSTM фіксує важливі функції з входів і зберігає цю інформацію протягом тривалого періоду часу. Рішення про видалення чи збереження інформації приймається на основі значень ваг, які були присвоєні інформації в процесі тренування.

Отже, модель LSTM дізнається, яку інформацію варто зберегти чи видалити. Взагалі, модель LSTM складається з трьох вузлів: вузел забуття, який забуває інформацію, вхідного та вихідного шлюзів. Вузел забуття приймає рішення про збереження / видалення існуючої інформації, вхідний вузел визначає ступінь додавання нової інформації до пам'яті, а вихідний вузел контролює, чи сприяє наявне значення в клітині виводу.

Вузел забуття. Зазвичай для цього етапу використовується сигмовидна функція для прийняття рішення про те, яку інформацію потрібно видалити з пам'яті LSTM. Результатом роботи цих воріт є значення від 0 до 1, де 0 означає повне позбавлення від вивченого значення, а 1 означає збереження цілого значення.

Вхідний вузел. Цей вузел приймає рішення про те, чи буде нова інформація додаватися в пам'ять LSTM. Ці ворота складаються з двох шарів: сигмоподібного шару та шару  $\tanh$ . Сигмоподібний шар вирішує, які значення потрібно оновити, а шар  $\tanh$  створює вектор нових значень-кандидатів, які будуть додані до пам'яті LSTM. Поєднання цих двох рівнів забезпечує оновлення для пам'яті LSTM, в якому

поточне значення забувається за допомогою шару забуття через множення старого значення з подальшим додаванням нового значення-кандидата.

Вихідний вузел. Цей вузел спочатку використовує сигмоподібний шар, щоб прийняти рішення про те, яка частина пам'яті LSTM вносить вихід. Потім він виконує нелінійну функцію  $\tanh$  для відображення значень між  $-1$  і  $1$ . Нарешті, результат множиться на вихід сигмоїдного шару.

BiLSTM [25] є продовженням описаних моделей LSTM, в яких два LSTM застосовуються до вхідних даних. У першому раунді LSTM застосовується до вхідної послідовності (тобто прямого шару). У другому раунді зворотна форма вхідної послідовності подається в модель LSTM (тобто зворотний шар). Застосування LSTM двічі призводить до поліпшення вивчення довгострокових залежностей і, отже, покращує точність моделі [3].

GRU – це нове покоління рекурентних нейронних мереж і дуже схоже на LSTM. GRU позбувся стану клітини та використав прихований стан для передачі інформації. Він також має лише два вузли: вузел скидання та вузел оновлення.

Вузел оновлення діє подібно до вхідного вузла та вузла забуття LSTM. Він вирішує, яку інформацію викидати та яку нову інформацію додавати. Вузел скидання – це ще один вузел, який використовується для вирішення, скільки минулої інформації забути. GRU має менше тензорних операцій, тому вони трохи швидше тренуються, ніж LSTM.

Двонаправлений керований рекурентний блок (BiGRU) виглядає точно так само, як його однонаправлений аналог. Різниця полягає в тому, що ворота пов'язані не просто з минулим, а й з майбутнім.

Основна архітектура CNN спочатку була розроблена для роботи над зображеннями. Згорткові шари призначені для вилучення шаблонів об'єктів з низького рівня на високий рівень. У класифікації тексту також використовувались мережі CNN. Пов'язану роботу в такому домені можна розділити на роботу над функціями рівня символів, функціями рівня слова або обома рівнями для функцій.

### 1.3 Постановка задачі

Під час аналізу предметної галузі було сформовано вимоги до подальшої роботи. Були виявлені основні напрями та рішення для проведення порівняння.

Основним завданням є пошук універсального підходу для роботи з різними типами текстових даних, який дає найбільший вигаш в якості.

В роботі потрібно реалізувати різні варіації нейромережевих підходів, які використовуються для розпізнавання емоцій та для роботи з текстовими даними. Для об'єктивного порівняння потрібно розглянути роботу реалізованих способів на різних вхідних даних. Для цього потрібно зібрати набори тестових даних, які маркіровані за емоційною складовою та максимально наближуються до текстів в реальному житті. Після цього потрібно натренувати та оцінити якість моделей на тестових даних за допомогою обраних метрик.

Також для порівняння слід розробити гібридний підхід, який базується на суміщенні робіт двох незалежних нейронних мереж, що відповідатимуть за різні функції у навчанні. Треба реалізувати та оцінити якість моделей побудованих за розробленим підходом та порівняти зі звичайними моделями.

При реалізації нейромережевих підходів слід виконати наступні завдання:

- для навчання нейронної мережі треба провести збір даних;
- в якості нейронної мережі слід розробити та реалізувати мережу з архітектурою, що підходить для роботи з текстовими даними, яка буде здійснювати класифікацію за емоційною складовою в тексті;
- при реалізації архітектури в програмному коді слід використовувати фреймворк Tensorflow або Keras;
- при побудові архітектури слід використовувати технологію обробки текстових даних word embedding.

У результаті потрібно провести комплексну оцінку ефективності отриманих моделей за допомогою обраних метрик та провести порівняльний аналіз звичайних моделей та моделей побудованих за гібридним принципом.

## 2 ОПИС МЕТОДІВ ДОСЛІДЖЕНЬ

### 2.1 Набори даних

Для оцінки ефективності різних архітектур було використано набори даних, які були розмічені за емоційною складовою [16] з різних доменів (діалоги, твіти, блоги та анкети):

- Emoint – набір даних, що містить тексти із публікацій у Twitter; твіти розмічені краудсорсингом; розмітка демонструє концентрацію гніву, радості, смутку, страху [17]; містить сім тисяч твітів;

- ISEAR – набір даних, побудований на основі різних анкет [18]; він має сім класів: радість, страх, гнів, смуток, огиду, сором, провину і містить шість тисяч зразків;

- StackOverflow – набір даних, побудований на запитаннях та відповідях StackOverflow, був зібраний за допомогою краудсорсингу; містить 24К публікацій та 6 емоцій, таких як любов, страх, радість, смуток, гнів, здивування;

- CrowdFlower – набір даних Twitter, опублікований CrowdFlower; містить 30 тисяч твітів, 5 емоцій (нейтральність, щастя, сум, гнів, страх); твіти розмічені краудсорсингом [19];

- DailyDialogs – набір даних, що містить діалоги, позначені емоціями [20]; містить 13 тисяч діалогів, 100 тисяч промов, сім емоцій.

Набір даних Emoint містить понад сім тисяч зразків. Набір даних компілюється та розмічається за допомогою краудсорсингу та знаходиться у відкритому доступі. Кожен зразок був представлений текстовим повідомленням з мікроблогу Twitter, позначеною емоцією та показником інтенсивності цієї емоції. Індикатор інтенсивності показує, наскільки емоція виражена в даному зразку, його значення коливається від 0 до 1, де 0 – повна відсутність емоцій у вибірці, 1 – текст повністю складається з позначеної емоції. При детальному вивченні набору даних було вирішено використовувати зразки з інтенсивністю емоцій понад 0.6, тому що при інтенсивності менше 0.6 емоції погано представлені або відсутні. Очищений

набір даних EmoInt для експерименту містить 2245 зразків з розміткою на 4 класи емоцій.

Набір даних ISEAR містить 7666 зразків, кожен зразок – це текст відповідей, зібраних з різних анкет, та емоції, представленої у цьому тексті. Для експерименту використовується весь набір даних.

Набір даних StackOverflow містить 24 000 публікацій StackOverflow, позначених за 6 класами емоцій (любов, страх, радість, смуток, гнів, здивування) трьома оцінювачами. Після детального вивчення було виявлено, що більшість постів не мають чіткої розмітки – розмітка або відсутня, або мітка визначається лише одним оцінювачем. Після очищення набір даних містить 2929 зразків, які використовуються в подальшому експерименті.

Набір даних CrowdFlower містить 47288 зразків, які складаються з тексту повідомлень з Twitter та розмітки для 5 класів емоцій (нейтральний клас, щастя, сум, гнів, страх). Набір даних, очищений від немаркованих зразків, містить 47206 зразків.

Набір даних DailyDialogs містить 13118 діалогів, зібраних та розмічених краудсорсингом. Кожне повідомлення в діалогових вікнах позначено однією з 7 емоцій (0: відсутні емоції, 1: гнів, 2: огида, 3: страх, 4: щастя, 5: сум, 6: здивування). Для підготовки до експерименту діалоги були розділені на повідомлення з відповідними емоційними позначками – загалом 102968 позначених повідомлень. При вивченні отриманих даних було виявлено, що лише 17407 повідомлень мають емоційну позначку, всі інші повідомлення були позначені без емоцій.

Для експерименту використовувались лише повідомлення з емоційними мітками, як показано в таблиці 2.1.

Таблиця 2.1 – Класи емоцій в датасетах

	EmoInt	ISEAR	StackOverflow	CrowdFlower	DailyDialogs
Гнів	474	1096	882	1109	1120
Позитив	-	-	-	16297	12885

Кінець таблиці 2.1

	EmoInt	ISEAR	StackOverflow	CrowdFlower	DailyDialogs
Радість	562	1094	491	-	-
Кохання	-	-	1220	-	-
Смуток	513	1096	230	15938	1150
Подив	-	-	-	-	1823
Страх	696	1095	106	-	174
Провина	-	1093	-	-	-
Сором	-	1096	-	-	-
Ненависть	-	-	-	4301	-
Нейтральний	-	-	-	9643	-
Огида	-	-	-	-	353
Разом	2245	7666	2929	47206	17407

Набори даних були попередньо оброблені, де було використано нормалізацію та лематизацію [21]. Під час застосування нормалізації тексти були очищені від розділових знаків, а всі літери переведені на малі літери. Потім була проведена лемматизація – слова були зменшені до нормальної форми, а слова-зупинки видалені. Наступним етапом підготовки даних до навчання було підведення зразків даних до форми, яка була б зручною для використання їх як вхідних параметрів нейронної мережі.

Текст був векторизований із застосуванням бібліотеки Keras [22]. Кожне слово в тексті асоціювалось із числовим індексом у словнику, і всі зразки даних були представлені як вектори чисел. Кожен вектор доповнювався нулями до постійної довжини, щоб довжина тексту не впливала на остаточну здатність нейронної мережі узагальнювати.

## 2.2 Ембеддінги слів

В експерименті брали участь кілька типів ембеддінгів: ембеддінги GloVe, Word2vec, FastText та EWE. Для GloVe, Word2Vec, FastText було використано попередньо навчені ембеддінги слів:

- попередньо навчений ембеддінг GloVe містить 42В лексеми, 1.9 мільйони словникових запасів унікальних лексем та векторне представлення кожного токена – 300-вимірні вектори англійських слів;
- для підходу Word2vec було використано попередньо навчений ембеддінг, що було натреновано на корпусі новин Google із 3 мільярдами лексем, векторна модель слів має 3 мільйони 300-вимірних векторів англійських слів;
- для підходу FastText було використано попередньо навчені вектори слів, які тренувались за допомогою Common Crawl та Wikipedia; моделі навчались із використанням безперервного мішка слів із вагами позицій, із n-грамами символів довжиною 5, та вікном розміру 5; розмір векторів англійських слів складався з 300 вимірів.

Emotion word embedding (EWE) був отриманий в результаті тренування шару ембеддінгу нейронної мережі. Для цього була створена нейронна мережа, що складається із шару ембеддінгу та щільного шару, як показано на рисунку 2.1. Шар ембеддінгу ініціалізується нульовою матрицею з увімкненою функцією навчання.

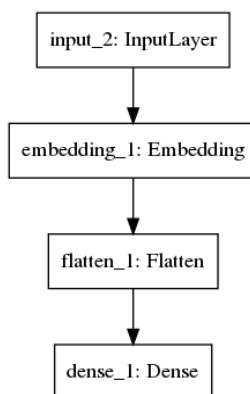


Рисунок 2.1 – Архітектура мережі для навчання EWE

Для навчання було створено універсальний зразок тренувальних даних, який є складом усіх наборів даних, щоб дані не перетиналися з тестовими зразками в майбутньому. Отриманий шар являє собою щільний шар з функцією активації Softmax, кількість нейронів дорівнює загальній кількості класів у всіх зразках – 12. Для навчання був використаний алгоритм зворотного розповсюдження, а як функцію втрат використовуємо категоріальну перехресну ентропію.

Гіперпараметри були налаштовані для кращої роботи тренованого ембедінгу. В таблиці 2.2 зображено значення гіперпараметрів, які були використані у порівнянні для подальшого навчання EWE.

Таблиця 2.2 – Підбір гіперпараметрів для навчання EWE

Гіперпараметр	Значення 1	Значення 2	Значення 3
Оптимізатор	Adam	Adadelata	RMSprop
Розмір батчу	50	100	300
Швидкість навчання	0.001	0.1	10
Кількість епох	15	10	5
Функція втрат	Категоріальна перехресна ентропія	-	-
Функція активації	Softmax	-	-

Було визначено, який гіперпараметр кращий, тренуючи та тестуючи одну і ту ж модель з різними значеннями відповідного гіперпараметра. Для тестування був використаний показник оцінки F1. В результаті було обрано модель, треновану за допомогою оптимізатора Adam (F1-score – 0.57, в Adadelata – 0.53, в RMSprop – 0.54). Кількість семплів, які подаються на вхід нейронної мережі, була обрана рівною 100, також було обрано тренування зі швидкістю навчання 0,001 та тривалістю 5 епох.

## 2.3 Метрики

В якості метрики була використана F-міра. Її обчислюють через влучність та повноту тесту, де влучність є числом правильно визначених позитивних результатів, поділеним на число всіх позитивних результатів, включно з визначеними неправильно, а повнота є числом правильно визначених позитивних результатів, поділеним на число всіх зразків, які повинно було бути визначено як позитивні. Також для оцінки ефективності моделей використовували метрику точності.

Також була використана зважена модифікація оцінки F1. Загальна оцінка рахується згідно з формулою (1).

$$F1 = F1_{class1} * W_1 + F1_{class2} * W_2 + \dots + F1_{classN} * W_N, \quad (1)$$

де  $F1_{classi}$  – значення F1 для класу  $i$ ;

$W_i$  – вагові коефіцієнти.

Вона обчислює оцінку F1 для кожного класу незалежно, але при їх додаванні використовуються вагові коефіцієнти  $W_i$ , які залежать від кількості справжніх міток кожного класу.

## 2.4 Моделі

Для порівняльного аналізу було проведено ряд експериментів на різних моделях для вирішення проблеми класифікації емоцій на основі текстових даних.

У роботі розглядалися такі моделі:

– BiLSTM;

- CNN;
- BiGRU;
- гібридна модель.

Для того, щоб збільшити навчальну здатність до архітектур порівнянних моделей було додано додаткові функціональні шари. Рівень відсіву був включений перед вихідним шаром, що допомагає поліпшити якість навчання моделей [23]. Основною метою методу відсіву є запобігання переобладнанню, і він зазвичай використовується для регуляризації штучних нейронних мереж. Суть методу полягає в тому, що під час навчання з підмережі нейронної мережі випадковим чином розрізняють та проводять навчання для підмережі. Навчальна підмережа забезпечується виключенням нейронів із початкової мережі з імовірністю  $p$ , і ймовірність того, що нейрон буде в мережі, рівний  $1-p$ , а виключені нейрони не сприяють жодним етапам алгоритму зворотного розповсюдження. Тож виключення принаймні одного з нейронів означає вивчення нової нейронної мережі. Було вирішено зробити вихідний шар повним шаром з  $N$  нейронами, які відповідають міткам класифікованих емоцій.

Наступним, як функцію активації нейронів, було обрано функцію Softmax [24]. Категорична перехресна ентропія застосовується для класифікації однієї мітки, яка дорівнює одній категорії для кожної точки даних. Отже, вибірка даних може бути пов'язана виключно з одним класом [25]. Категорична перехресна ентропія використовується з функцією Softmax. Під час порівняння було використовували однакові налаштування гіперпараметрів для всіх моделей, як показано в таблиці 2.3.

Таблиця 2.3 – Гіперпараметри для навчання

Гіперпараметр	Значення
Оптимізатор	Adam
Розмір батчу	150
Швидкість навчання	0.001

Кінець таблиці 2.3

Гіперпараметр	Значення
Кількість епох	0.2
Функція втрат	Категоріальна перехресна ентропія

В якості оптимізатора було вирішено використовувати алгоритм оптимізації Adam. Adam відрізняється від класичного стохастичного градієнтного спуску. Стохастичний градієнтний спуск підтримує єдину швидкість навчання для всіх оновлень ваги, і швидкість не змінюється під час навчання. Швидкість навчання встановлюється для кожної ваги мережі (параметра) і окремо адаптується в міру розгортання навчання.

Всі набори даних були розділені на тренувальвальний та тестовий датасети у пропорції 80/20, також для валідації кроків був використаний розділений тренувальний та перевірочний датасет у тій же пропорції для кожного з кроків. Для проведення експерименту в роботі було розроблено архітектури для кожної із порівнюваних нейронних мереж.

### 3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

#### 3.1 Архітектура BiLSTM

Першим методом розглядався BiLSTM. Для цього було створено архітектуру нейронної мережі, як показано на рисунку 3.1. Шари – це ембедінги, двонаправлена довготривала короткочасна пам'ять (двонаправлений LSTM), пулінг, випадаючі шари та два щільних шари.

Спочатку вхідні дані проходять через шар ембедінгу. Для порівняння роботи цієї архітектури з різними шарами ембедінгу моделі тренуються для кожного типу попередньо навченого ембедінгу: GloVe, Word2Vec, FastText, EWE. Наступний шар – BiLSTM. Також було використано найпопулярніший метод для пулінгу – функцію max, яка була виражена у шарі GlobalMaxPooling [26].

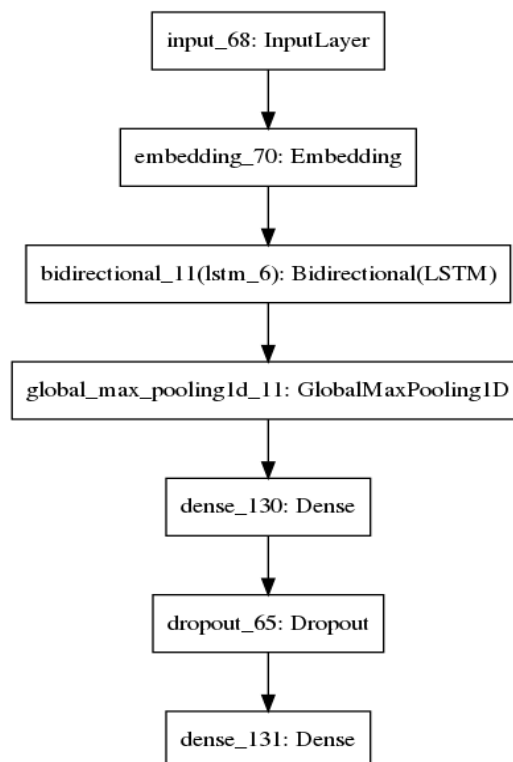


Рисунок 3.1 – Архітектура мережі BiLSTM

Було навчено та протестовано модель для кожного типу досліджуваного ембедінгу та для кожного типу наборів даних. Спочатку була навчена модель із

ембедінгом GloVe. Для тренування та випробування було заморожено її вбудовуючий шар, щоб не погіршити можливості узагальнення моделі. Для отримання результатів модель була перевірена на тестовому датасеті з використанням 2 метрик – точність та зважена F1-міра, як показано в таблиці 3.1.

Таблиця 3.1 – Результати тестування BiLSTM з Glove

Dataset	Accuracy	Weighted F1-score
EmoInt	0.85	0.86
ISEAR	0.55	0.56
StackOverflow	0.71	0.76
CrowdFlower	0.62	0.61
DailyDialogs	0.85	0.85

Максимальний бал F1 становить 0.86 для DailyDialogs, мінімальний бал – 0.56 – для ISEAR. Найбільш збалансований розподіл прогнозів за класами належить EmoInt – кожен із класів має щонайменше 82% справжніх позитивних результатів. Матриця невідповідностей для EmoInt представлена на рисунку 3.2.

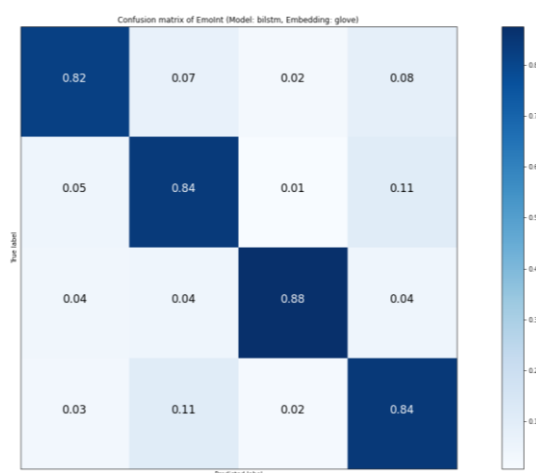


Рисунок 3.2 – Матриця невідповідностей EmoInt

Наступним було протестовано модель разом з ембедінгом EWE. Як і раніше, було відключено навчання для цього рівня та поєднано його з архітектурою мережі BiLSTM. Результати показані в таблиці 3.2.

Таблиця 3.2 – Результати тестування BiLSTM з EWE

Dataset	Accuracy	Weighted F1-score
EmoInt	0.79	0.79
ISEAR	0.53	0.53
StackOverflow	0.70	0.72
CrowdFlower	0.55	0.54
DailyDialogs	0.82	0.83

Найбільш збалансований розподіл прогнозів за класами належить EmoInt. Максимальний бал F1 становить 0.83 для DailyDialogs, мінімальний бал – 0.53 для ISEAR. Більшість істино позитивних результатів у прогнозах DailyDialogs належать до класу «щасливих» – 94% істинно позитивних прогнозів. Прогнози EmoInt варіюються від 71% до 85% точності серед усіх класів. Прогнози для класів Crowdfower та ISEAR варіюються від 43% до 68%. Точність прогнозованих класів StackOverflow коливається від 40% до 89%.

Після ембедінгу EWE було побудовано та протестовано архітектуру BiLSTM із ембедінгом FastText, яка була натренована згідно з гіперпараметрами, які були обрані на початку роботи. Результати тестування натренованої моделі показано в таблиці 3.3.

Таблиця 3.3 – Результати тестування BiLSTM з FastText

Dataset	Accuracy	Weighted F1-score
EmoInt	0.42	0.46
ISEAR	0.39	0.40
StackOverflow	0.54	0.64
CrowdFlower	0.61	0.62
DailyDialogs	0.80	0.83

Максимальний бал F1 – 0.83 для DailyDialogs, мінімальний – 0.40 для ISEAR. Найбільш збалансований розподіл передбачень за класами належить набору даних CrowdFlower. Точність передбачуваних класів коливається від 65% до 69%. Для DailyDialogs розподіл точності між класами майже подібний, як у випадку ембеддінгу EWE.

Наступним кроком було навчено модель із ембеддінгом Word2Vec та досліджено її ефективність, використовуючи зібрані датасети. Результати показані в таблиці 3.4.

Таблиця 3.4 – Результати тестування BiLSTM з Word2Vec

Dataset	Accuracy	Weighted F1-score
EmoInt	0.74	0.75
ISEAR	0.55	0.57
StackOverflow	0.66	0.76
CrowdFlower	0.62	0.63
DailyDialogs	0.83	0.84

Найбільш збалансований розподіл передбачень за класами належить набору даних CrowdFlower. Максимальний бал F1 становить 0.84 для DailyDialogs, мінімальний бал – 0.57 для ISEAR.

Підводячи підсумок, була порівняна ефективність моделі BiLSTM для кожного типу ембеддінгу за допомогою зваженої метрики F1-балів. Результати показані в таблиці 3.5.

Таблиця 3.5 – Загальні результати тестування BiLSTM

Dataset	GloVe	FastText	Word2Vec	Sentiment
EmoInt	0.85	0.46	0.75	0.79
ISEAR	0.56	0.40	0.57	0.53
StackOverflow	0.76	0.64	0.76	0.72

Кінець таблиці 3.5

Dataset	GloVe	FastText	Word2Vec	Sentiment
CrowdFlower	0.61	0.62	0.63	0.54
DailyDialogs	0.86	0.83	0.84	0.83

Найкращу продуктивність показали GloVe та Word2Vec, але Word2Vec показав кращий результат для більш збалансованих наборів даних – CrowdFlower та ISEAR (обидва не мають явних викидів даних), що означає кращі можливості узагальнення.

### 3.2 Архітектура CNN

Модель CNN складається з вбудованого шару і шару згортки, прихованих шарів, шару об'єднання та відсіву. Архітектура мережі представлена на рисунку 3.3.

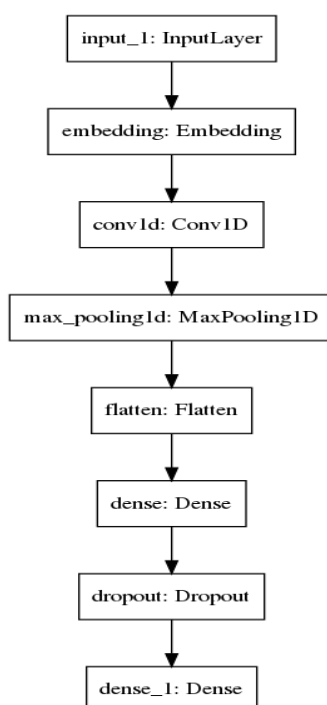


Рисунок 3.3 – Архітектура моделі CNN

Задля покращення навчальної здібності моделі, в її архітектуру був доданий шар пулінгу [27]. Модель використовує останній прихований стан для розпізнавання емоцій.

Збудовану модель було навчено та протестовано для кожного типу досліджуваних вбудовувань та для кожного типу наборів даних.

Спочатку було натреновано модель із ембедінгом GloVe, який попередньо було заморожено. Результати роботи моделі на тестових даних показано в таблиці 3.6.

Таблиця 3.6 – Результати тестування CNN з GloVe

Dataset	Accuracy	Weighted F1-score
EmoInt	0.54	0.57
ISEAR	0.33	0.35
StackOverflow	0.59	0.66
CrowdFlower	0.55	0.55
DailyDialogs	0.80	0.83

Максимальний бал за метрикою F1 був для DailyDialogs – 0.83, мінімальний – 0.35 для ISEAR. Точність прогнозування класів EmoInt коливається в межах від 61% до 88%. Більшість прогнозів класів набору даних ISEAR мають точність менше 50%, зважена точність серед усіх класів – 33%. Лише 2 класи датасету StackOverflow мають точність, що перевищує 50% – любов (73%) та гнів (80%), це пов'язано з тим, що ці класи найбільше представлені в наборі даних. Точність прогнозованих класів CrowdFlower коливається від 41% (нейтральний клас) до 70% (клас «страх»).

Наступним ембедінгом, з яким було навчено модель, був ембедінг EWE, як показано в таблиці 3.7.

Таблиця 3.7 – Результати тестування CNN з EWE

Dataset	Accuracy	Weighted F1-score
EmoInt	0.78	0.78
ISEAR	0.50	0.50
StackOverflow	0.69	0.71
CrowdFlower	0.54	0.54
DailyDialogs	0.81	0.81

Максимальний бал F1 становив 0.81, мінімальний – 0.50. Для EmoInt усі класи мають точність понад 70%. Показник точності прогнозів ISEAR становить від 37% до 67% для кожного класу. CrowdFlower та DailyDialogs мають подібні результати, як GloVe. Прогнози StackOverflow для нейтрального класу мають кращу точність, ніж підхід GloVe – 80% та 73% відповідно.

Прогнози FastText мають менший показник зваженого F1, ніж підхід з ембедінгом EWE для всіх перевірених наборів даних, за винятком CrowdFlower – нейтральний клас має 38% точності, тоді як EWE має 30% для того ж класу, а щасливий клас має 70% проти 58% у підході з використанням EWE, як показано у таблиці 3.8.

Таблиця 3.8 – Результати тестування CNN з FastText

Dataset	Accuracy	Weighted F1-score
EmoInt	0.41	0.43
ISEAR	0.26	0.27
StackOverflow	0.56	0.64
CrowdFlower	0.57	0.57
DailyDialogs	0.76	0.80

Мінімальний показник зваженого F1 для Word2Vec було отримано для набору даних ISEAR. Його передбачувана точність для кожного класу коливається

від 18% ("почуття провини") до 56% ("огида"). Точність прогнозування емоцій коливається від 22% (злість) до 75% (сум). Для інших наборів даних відмінності незначні, як показано в таблиці 3.9.

Таблиця 3.9 – Результати тестування CNN з Word2Vec

Dataset	Accuracy	Weighted F1-score
EmoInt	0.48	0.53
ISEAR	0.36	0.36
StackOverflow	0.58	0.64
CrowdFlower	0.56	0.56
DailyDialogs	0.79	0.82

Порівнюючи результати експериментів для кожного типу ембедінгу було знайдено, що архітектура CNN з EWE ембедінгом дає кращі результати, ніж інші методи ембедінгів, оскільки воно дає значний виграш для розріджених наборів даних, таких як EmoInt та StackOverflow, і забезпечує кращу продуктивність для збалансованого набору даних ISEAR. Результати тестування наведені в таблиці 3.10.

Таблиця 3.10 – Загальні результати тестування CNN

Dataset	GloVe	FastText	Word2Vec	Sentiment
EmoInt	0.57	0.43	0.53	0.78
ISEAR	0.35	0.27	0.36	0.50
StackOverflow	0.66	0.64	0.64	0.71
CrowdFlower	0.55	0.57	0.56	0.54
DailyDialogs	0.83	0.80	0.82	0.81

Також було знайдено, що згідно з тестуванням, для CrowdFlower та DailyDialogs всі ембедінги показали подібні результати.

### 3.3 Архітектура BiGRU

Наступні експерименти були присвячені моделі з BiGRU. Архітектура нейронної мережі була створена, як показано на рисунку 3.4. Взаємопов'язані шари архітектури складаються із ембедінгів, двонаправленого GRU, шару пулінгу, шарів дропауту і двох щільних шарів [27].

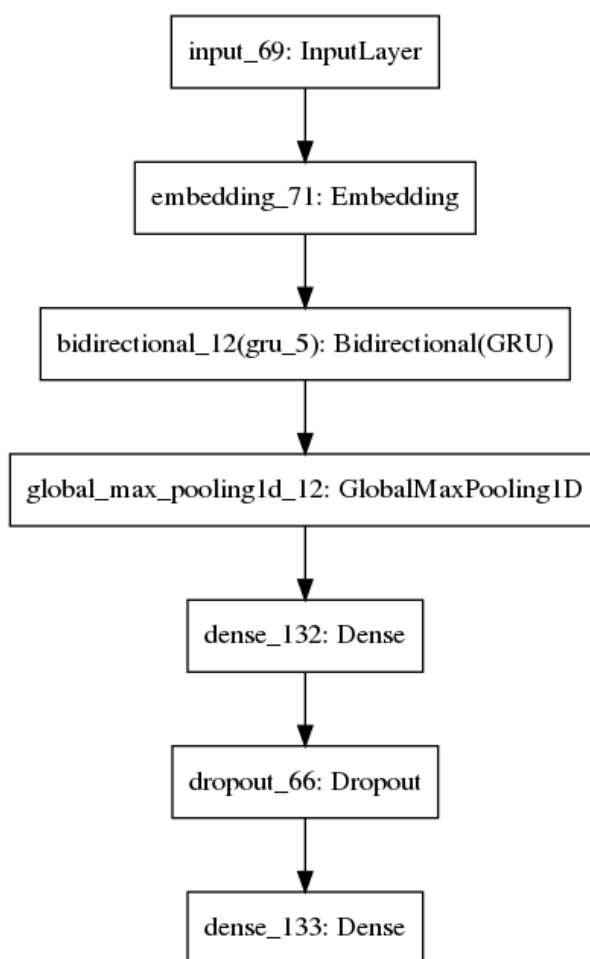


Рисунок 3.4 – Архітектура моделі BiGRU

Підхід з BiGRU та GloVe має максимальний показник зваженого F1 для EmoInt – 0.90, як показано в таблиці 3.11. Точність прогнозування коливається від 76% до 98%. Мінімальний бал F1 для ISEAR – 0.53: точність між 39% і 74%.

Таблиця 3.11 – Результати тестування BiGRU з GloVe

Dataset	Accuracy	Weighted F1-score
EmoInt	0.90	0.90
ISEAR	0.54	0.53
StackOverflow	0.70	0.78
CrowdFlower	0.61	0.62
DailyDialogs	0.84	0.85

Підхід із Emotion word embedding показує однакову або гіршу ефективність для кожного типу набору даних як для усередненої точності, так і для метрики зваженого F1. Максимальний зважений F1 – для EmoInt (0.85), мінімальний – для збалансованого набору даних ISEAR (0.53) та CrowdFlower (0.55), як показано в таблиці 3.12.

Таблиця 3.12 – Результати тестування BiGRU з EWE

Dataset	Accuracy	Weighted F1-score
EmoInt	0.85	0.85
ISEAR	0.54	0.53
StackOverflow	0.70	0.71
CrowdFlower	0.55	0.55
DailyDialogs	0.82	0.83

В експерименті підходу BiGRU з FastText, максимальний показник зваженого F1, належить до набору даних DailyDialogs і мінімальний як для наборів даних EmoInt, так і для ISEAR. Цей підхід показує найгіршу ефективність для набору даних EmoInt – модель перенавчається [28] на тренувальних даних для класу «радість» і більшість немічених тестових даних були позначені як «радість». Результати тестування зображені у таблиці 3.13.

Таблиця 3.13 – Результати тестування BiGRU з FastText

Dataset	Accuracy	Weighted F1-score
EmoInt	0.37	0.44
ISEAR	0.43	0.44
StackOverflow	0.62	0.71
CrowdFlower	0.61	0.62
DailyDialogs	0.82	0.85

Згідно з результатами тестування був проведений детальний аналіз розподілу класів для датасету EmoInt. Більш детальна інформація наведена на матриці невідповідностей на рисунку 3.5.

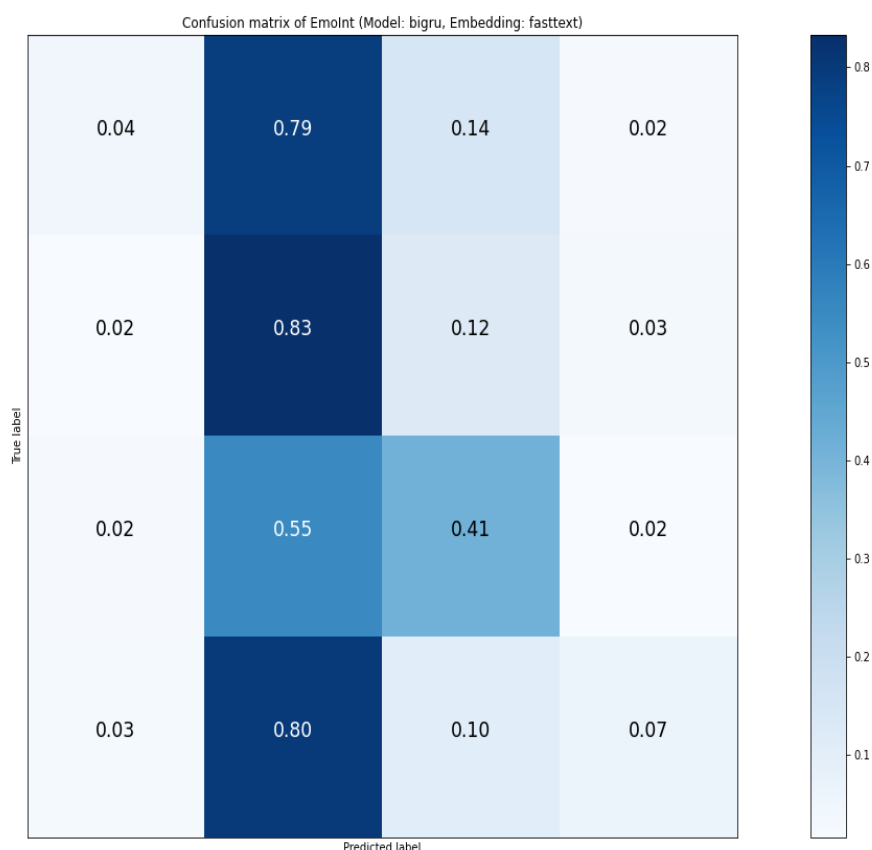


Рисунок 3.5 – Матриця невідповідностей EmoInt для BiGRU з FastText

Як можна зробити висновок, BiGRU з FastText демонструє кращу продуктивність, ніж підход з Emotion word embedding, та подібну продуктивність

із підходом GloVe на великому збалансованому наборі даних CrowdFlower. Але зважений F1 набагато гірший для малих наборів даних EmoInt та ISEAR, ніж показали відповідні підходи. Можливість передбачення моделі для StackOverflow, яку було враховано за зібраними показниками, показує, що протестована модель гірше виявила результати для набору даних, ніж GloVe. Підводячи підсумок, BiGRU з FastText працює краще для більших наборів даних.

Для Word2Vec мінімальний зважений F1-бал належить набору даних ISEAR – варіація точності прогнозованих класів становить від 12% до 54%, як показано в таблиці 3.14.

Таблиця 3.14 – Результати тестування BiGRU з Word2Vec

Dataset	Accuracy	Weighted F1-score
EmoInt	0.38	0.51
ISEAR	0.24	0.29
StackOverflow	0.50	0.60
CrowdFlower	0.58	0.60
DailyDialogs	0.79	0.84

Проаналізувавши всі можливі варіанти ембедінгів та порівнявши результати, було зроблено висновок, що найкращим підходом для BiGRU є використання GloVe. Порівняльні результати тестування зображено у таблиці 3.15.

Таблиця 3.15 – Загальні результати тестування BiGRU

Dataset	GloVe	FastText	Word2Vec	Sentiment
EmoInt	0.90	0.44	0.51	0.85
ISEAR	0.53	0.44	0.29	0.53
StackOverflow	0.78	0.71	0.60	0.71
CrowdFlower	0.62	0.62	0.60	0.55
DailyDialogs	0.85	0.85	0.84	0.83

BiGRU у поєднанні з GloVe показав найкращу продуктивність для кожного типу наборів даних, як показано в таблиці 3.15. BiGRU з FastText демонструє хорошу продуктивність для наборів даних, які містять більше 10000 зразків – CrowdFlower та DailyDialogs, але для менших наборів даних він працює гірше, ніж із вбудованим GloVe.

### 3.4 Гібридний підхід

Після аналізу звичайних архітектур і отримання порівняльних результатів було прийнято рішення розробити гібридний метод побудови моделі. Архітектура такої моделі складається з двох субмоделей, які відповідають за різні функції. Будова моделі зображена на рисунку 3.6.

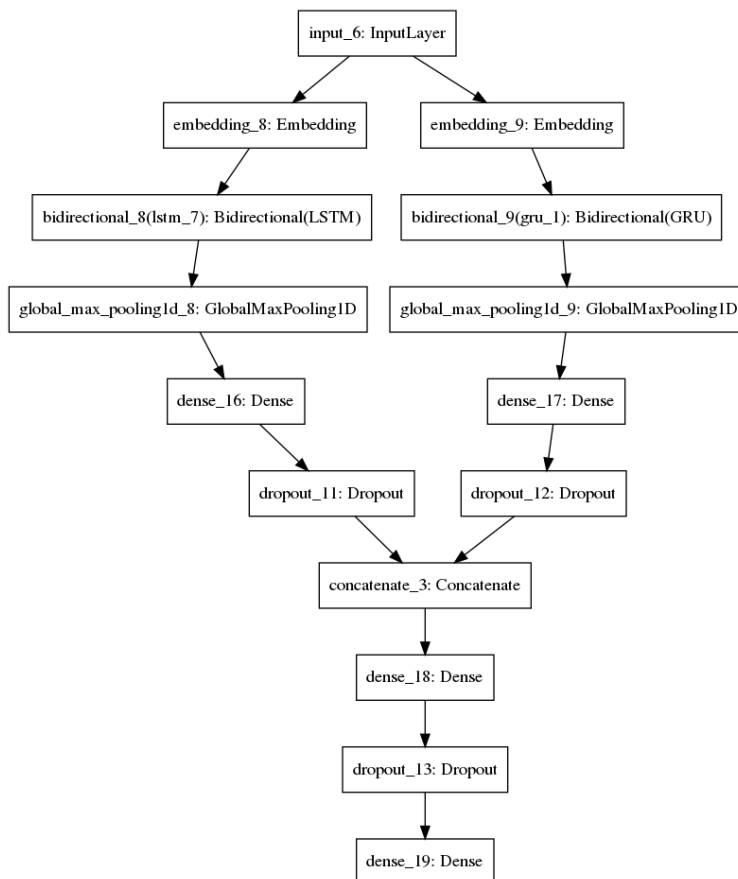


Рисунок 3.6 – Архітектура гібридної моделі

Перша субмодель відповідає за виділення сентиментної складової вхідного тексту: вона складається з EWE ембедінгу, яке допомагає знайти емоційний зв'язок між словами, шару рекурентної нейронної мережі та допоміжних шарів таких як пулінг, дропаут. Друга модель являє собою нейронну мережу, головне завдання якої знайти семантичний зв'язок поміж вхідними словами. Вона побудована на базі нейронних мереж, які були протестовані. У якості було субмоделі було обрано архітектури BiGRU з GloVe та BiLSTM з Word2Vec, які продемонстрували найкращі результати на тестових виборках.

Після того, як дані пройшли через обидві субмоделі, їх вихід конкатенується та пропускається через слой дропауту.

Архітектура, яка була побудована на базі BiGRU з GloVe, була протестована на тестових виборках, що використовувалися раніше. Результати тестування можна побачити в таблиці 3.16.

Таблиця 3.16 – Результати тестування гібридної архітектури з BiGRU та GloVe

Dataset	Accuracy	Weighted F1-score
EmoInt	0.91	0.90
ISEAR	0.57	0.56
StackOverflow	0.70	0.77
CrowdFlower	0.64	0.65
DailyDialogs	0.83	0.84

Згідно з результатами гібридна модель дала основний приріст на збалансованих датасетах таких як ISEAR та CrowdFlower. Для ISEAR точність покращилася з 54% до 57%, зважений F1 покращився з 0.53 до 0.56. Для CrowdFlower точність зросла з 61% до 64%, а показник F1 з 0.62 теж до 0.65. Тестування розріджених датасетів не дало значного приросту. В таблиці 3.17

зображено порівняння метрик зваженого F1 звичайного підходу BiGRU з GloVe та гібридного.

Таблиця 3.17 – Порівняння результатів тестування базової та гібридної архітектури

Dataset	BiGRU + GloVe	Hybrid BiGRU + GloVe
EmoInt	0.90	0.90
ISEAR	0.53	0.56
StackOverflow	0.78	0.77
CrowdFlower	0.62	0.65
DailyDialogs	0.85	0.84

Наступним було проведено тестування моделі з субмоделю, архітектура якої базувалася на BiLSTM з Word2Vec. Результати наведено у таблиці 3.18.

Таблиця 3.18 – Результати тестування гібридної архітектури з BiLSTM та Word2Vec

Dataset	Accuracy	Weighted F1-score
EmoInt	0.75	0.75
ISEAR	0.57	0.60
StackOverflow	0.65	0.75
CrowdFlower	0.63	0.65
DailyDialogs	0.82	0.84

Як і у гібридної архітектури показники метрик для збалансованих датасетів покращились. Метрика зваженого F1 зросла з 0.57 до 0.60 для ISEAR та з 0.63 до 0.65 для CrowdFlower.

Таблиця 3.19 – Порівняння результатів тестування базової та гібридної архітектури BiLSTM

Dataset	BiLSTM + Word2Vec	Hybrid BiLSTM + Word2Vec
EmoInt	0.75	0.75
ISEAR	0.57	0.60
StackOverflow	0.76	0.75
CrowdFlower	0.63	0.65
DailyDialogs	0.84	0.84

У таблиці 3.19 наведено порівняння результатів випробувань для підходу з BiLSTM з Word2Vec та гібридною архітектурою, яка базується на цьому підході.

### 3.5 Комплексна оцінка ефективності моделей

Для проведення остаточного аналізу було зібрано всі результати та порівняно їх між собою для кожного набору даних, як показано в таблиці 3.20. Набори даних: E – EmoInt, I – ISEAR, S – StackOverflow, D – DailyDialogs. Було використано зважений F1 як метрику представлення продуктивності моделі.

Таблиця 3.20 – Загальне порівняння усіх результатів

Dataset	E	I	S	C	D
BiLSTM + GloVe	0.85	0.56	0.76	0.61	0.86
BiLSTM + FastText	0.46	0.40	0.64	0.62	0.83
BiLSTM + Word2Vec	0.75	0.57	0.76	0.63	0.84
BiLSTM + Sentiment	0.79	0.53	0.72	0.54	0.83
CNN + GloVe	0.57	0.35	0.66	0.55	0.83
CNN + FastText	0.43	0.27	0.64	0.57	0.80

Кінець таблиці 3.20

Dataset	E	I	S	C	D
CNN + Word2Vec	0.53	0.36	0.64	0.56	0.82
CNN + Sentiment	0.78	0.50	0.71	0.54	0.81
BiGRU + GloVe	0.90	0.53	0.78	0.62	0.85
BiGRU + FastText	0.44	0.44	0.71	0.62	0.85
BiGRU + Word2Vec	0.51	0.29	0.60	0.60	0.84
BiGRU + Sentiment	0.85	0.53	0.71	0.55	0.83
Hybrid BiGRU + GloVe	0.75	0.60	0.75	0.65	0.84
Hybrid BiLSTM + Word2Vec	0.90	0.56	0.77	0.65	0.84

Було виявлено, що архітектура моделі з BiLSTM та Word2Vec як Embedding працює ефективніше для наборів даних із збалансованими класами – це означає, що набір даних має відносно достатню кількість зразків для кожного класу [29]. Також було виявлено, що для наборів даних з розрідженими класами, таких як EmoInt та StackOverflow, найкращу продуктивність показав BiGRU з архітектурою GloVe.

В результаті тестування гібридного підходу було виявлено, що використання додаткової субмоделі, яка обробляє лише емоціональну складову текстових даних, підвищує показники для збалансованих датасетів.

## ВИСНОВКИ

В ході роботи було проаналізовано проблеми та методи вирішення задачі розпізнавання емоцій за текстом, використовуючи технології поглибленого вивчення. В ході аналізу предметної області було проведено аналіз існуючих рішень. Більшість з існуючих рішень спеціалізовані під конкретні вхідні дані і не характеризують їх продуктивність на реальних текстових даних.

Для заданої теми було проведено аналіз існуючих методів для розпізнавання емоцій. Загалом всі методи базуються на використанні методів глибокого вивчення та нейронних мереж. Серед методів найбільш часто використовуються підходи пов'язані з рекурентними нейронними мережами та векторним уявленням слів – вкладаннями. При аналізі моделей найкращий результат досягли моделі BiLSTM з Word2Vec та BiGRU з GloVe.

Також було розроблено гібридний підхід, який базується на суміщенні робіт двох незалежних нейронних мереж, що відповідають за різні функції у навчанні. Було реалізовано та оцінено якість моделей побудованих за розробленим підходом та порівняно зі звичайними моделями. Завдяки використанню гібридного підходу було покращено результати звичайних архітектур, які було винайдено та протестовано в роботі.

Таким чином було зроблено висновок про поліпшення роботи нейронних мереж за допомогою додаткової архітектури, а також про найкращі поєднання підходів для вирішення задачі розпізнавання емоційної складової за текстовими даними.

Як результат, було актуалізовано ряд проблем та їх рішень, але все одно, існує ряд проблем, які важко вирішити на даний момент. В результаті роботи було обрано підходи для подальшого аналізу та перевірки їх здатності розпізнавати емоції за текстом.

## ПЕРЕЛІК ДжЕРЕЛ ПОСИЛАННЯ

1. Валаа Медхат, Ахмед Хасан і Хода Кораші. «Алгоритми та додатки аналізу настроїв: опитування». *Ain Shams Engineering Journal*, випуск 5, с1093-1113, 2014
2. Goodfellow I, Bengio Y, Courville A (2016) Глибоке навчання. MIT Press, Кембридж
3. Meisheri H, Dey L (2018) Дослідження TCS у завданні SemEval-2018 завдання 1: вивчення надійних уявлень із використанням архітектури багато уваги. В: Матеріали 12-го міжнародного практикуму з семантичного оцінювання. Асоціація обчислювальної лінгвістики, с. 291–299
4. Eisner B, Rocktäschel T, Augenstein I, Bosnjak M, Riedel S (2016) emoji2vec: вивчення подань смайлів з їх опису. В: Матеріали четвертого міжнародного семінару з обробки природних мов для соціальних мереж. Асоціація обчислювальної лінгвістики, с. 48–54
5. Pennington J, Socher R, Manning C (2014) Glove: глобальні вектори для подання слів. В: Матеріали конференції 2014 року з емпіричних методів в обробці природних мов (EMNLP). Асоціація обчислювальної лінгвістики, с. 1532–1543
6. Graves A, Schmidhuber J (2005) Структурна класифікація фонем з двонаправленим lstm та іншими архітектурами нейронних мереж. *Нейронна мережа* 18 (5): 602–610
7. Bahdanau D, Cho K, Bengio Y (2014) Нейронний машинний переклад шляхом спільного навчання вирівнюванню та перекладу. arXiv: 1409.0473
8. Wang Y, Feng S, Wang D, Yu G, Zhang Y (2016) Багатозначна класифікація китайських емоцій мікроблогів за допомогою згорткової нейронної мережі. У: Li F, Shim K, Zheng K, Liu G (eds) Веб-технології та програми: APWeb 2016, том 9931. Конспекти лекцій з інформатики. Спрінгер, Чам, с. 567–580
9. Quan C, Ren F (2010) Корпус емоційних блогів для аналізу емоційних виразів китайською мовою. *Мова обчислювальної мови* 24 (4): 726–749

10. Wang M, Liu M, Feng S, Wang D, Zhang Y (2014) Новий відкалібрований метод класифікації для виявлення множинних емоцій у китайських мікроблогах. У: Zong C, Nie JY, Zhao D, Feng Y (eds) Обробка природних мов та китайські обчислення. Спрінгер, Берлін, с. 238–250
11. Seyeditabari A, Tabari N, Gholizadeh S, Zadrozny W (2019) Виявлення емоцій у тексті: фокус на прихованому поданні. препринт arXiv arXiv: 1907.09369
12. Speer R, Chin J, Havasi C (2017) Conceptnet 5.5: відкритий багатомовний графік загальних знань. В: Матеріали тридцять першої конференції AAAI зі штучного інтелекту, AAAI'17. AAAI Press, с. 4444–4451
13. Міколов Т, Могила Е, Бояновський П, Пурш С, Джоулін А (2018) Досягнення в підготовці до розподілених подань слів. В: Матеріали одинадцятої міжнародної конференції з мовних ресурсів та оцінки (LREC-2018). Європейська асоціація мовних ресурсів (ELRA), Міядзакі
14. T. Mikolov, A. Deoras, S. Kombrink, L. Burget, J. Černocký, Empirical evaluation and combination of advanced language modeling techniques, in: Twelfth Annual Conference of the International Speech Communication Association, 2011.
15. Agrawal, A. An, M. Papagelis, Learning emotion-enriched word representations, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 950-961.
16. L.A.M. Bostan, R. Klinger, An Analysis of Annotated Corpora for Emotion Classification in Text, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 2104-2119.
17. S.M. Mohammad, F. Bravo-Marquez, Emotion intensities in tweets, 2017.
18. K.R. Scherer, H.G. Wallbott, Evidence for universality and cultural variation of differential emotion response patterning, Journal of personality and social psychology, 66(2), 1994, p.310.
19. S. Mohammad, S. Kiritchenko, Understanding emotions: A dataset of tweets to study interactions between affect categories, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), 2018.

20. Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu, Dailydialog: A manually labelled multi-turn dialogue dataset, 2017.
21. Eija, Word Normalization and Decompounding in Mono- and Bilingual IR//Information Retrieval, volume 9, 2006, Issue 3. pp. 249–271.
22. N. Ketkar, Introduction to Keras, in: Deep Learning with Python, Apress, Berkeley, CA, 2017, pp. 97-111.
23. G. Hinton, Improving neural networks by preventing co-adaptation of feature detectors, 2012, p. 18.
24. G. Bolin, P. Lacra, On the Properties of the Softmax Function with Application in Game Theory and Reinforcement Learning, 2017, p. 10.
25. K. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012, pp. 57-571.
26. B. Graham, Fractional Max-Pooling, 2015, p. 10.
27. Afanasieva I.V., et al. Neural network approach for emotional recognition in text / I.V. Afanasieva, D.S. Nazarenko, N.V. Golian // Біоніка інтелекту, Харків: XHYPE, 2019. – 1(92). – С. 9-14.
28. Maksym Bekuzarov, Oleksandr Samantsov, Oksana Mazurova, Mariia Shirokopetleva. Neural Network Architecture Editor With Code Generation. Problem of Infocommunications. Science and Technology (PIC S&T'2020), Kharkiv, Ukraine.- 6-9 October 2020.
29. Maksym Shopynskyi, Nataliia Golian, Iryna Afanasieva. Long short-term memory model appliance for generating music compositions // 2020 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), 6-9 Oct. 2020, Kharkiv, Ukraine.