Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження методів маркування візуальних об'єктів текстом в задачах розпізнавання зображень
(тема)

Виконав:
студент 2 курсу, групи СШІм-19-2
Байкалов М.О.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва спеціалізації)

Керівник к.т.н. Шевченко О. Ю.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

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

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Байкалову Максиму Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів маркування візуальних об'єктів текстом в задачах розпізнавання зображень

затверджена наказом університету від 29.03.2021 р. № 390Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ _____ 2021 р.

3. Вихідні дані до роботи Науково-технічні публікації щодо дослідження та розробки у сфері комп'ютерного зору для вирішення задачі маркування зображень; відомі набори даних для вирішення поставленої задачі, а саме вибірки даних з Kaggle репозіторія та їх опис. Інтернет-джерела та література з вказаної теми .

4. Перелік питань, що потрібно опрацювати в роботі Мета роботи; аналіз предметної області і постановка задачі дослідження; аналіз характеристик когнітивних систем та сервісів; аналіз підходів до розпізнавання зображень та тексту; розробка методу маркування зображень текстом; експериментальна перевірка методу маркування.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Рисунок 1 – Шкала часу з найбільш значущими дослідженнями у сфері комп'ютерного зору; Рисунок 2 – Маркування ліній довільної форми; Рисунок 3 – Моделювання за допомогою зображувальних структур; Рисунок 4 – Приклад варіацій інтенсивності затінення; Рисунок 5 – Приклад роботи алгоритму оптичного потоку на основі інтенсивності; Рисунок 6 – Приклад фігури затінення; Рисунок 7 – Приклад виявлення обличчя за допомогою активних контурів; Рисунок 7 – Приклад сегментації регіону; Рисунок 8, 9 та 10 – Загальний вигляд архітектури багатошарового персептрону; Рисунок 11 – Загальний вигляд стандартної нейронної мережі; Рисунок 12 – Багатошарове представлення стандартної нейронної мережі; Рисунок 13 та 14 – Обробка послідовності векторів на виході системи за допомогою RNN; Рисунок 15 – Паралельна обробка послідовності векторів за допомогою RNN; Рисунок 16 – Принцип роботи системи, що розглядається; Рисунок 17 – Матриця оцінок відповідностей між реченням та зображенням.

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
Основна частина	к.т.н. Шевченко О. Ю.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	29.03.21	виконано
2	Аналіз характеристик когнітивних систем та сервісів	30.03.21-01.04.21	виконано
3	Аналіз підходів до розпізнавання образів	02.04.21-04.04.21	виконано
4	Аналіз методів обробки природних мов	05.04.21-07.04.21	виконано
5	Розробка метод маркування зображень текстом	08.04.21-12.04.21	виконано
6	Експериментальна перевірка методу маркування	13.04.21-15.04.21	виконано
7	Обробка і оформлення результатів	16.04.21-17.04.21	виконано
8	Оформлення пояснювальної записки	18.04.21	виконано
9	Нормоконтроль	19.04. 21	виконано
10	Попередній захист	14.05.2021	виконано
11	Захист перед ЕК		

Дата видачі завдання 29 березня 2021 р.

Студент _____
(підпис)

Керівник роботи _____ ____ к.т.н. Шевченко О. Ю.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Записка пояснювальна: 93 с., 27 рис., 2 дод., 31 джерело.

АРХІТЕКТУРА НЕЙРОННОЇ МЕРЕЖІ, ВИЯВЛЕННЯ ОБ'ЄКТІВ, ЗАДАЧА МАРКУВАННЯ ЗОБРАЖЕНЬ, КОГНІТИВНІ СИСТЕМИ, ОБРОБКА ПРИРОДНОЇ МОВИ.

Метою даної кваліфікаційної роботи полягає у поєднанні методів розпізнавання зображень та тексту для того, щоб промаркувати зображення реченнями, що відображають сенс цих зображень, що дає можливість побудувати когнітивний сервіс, який має змогу одночасно працювати і з зображеннями і з текстом.

Об'єкт дослідження – комбінований процес розпізнавання зображень та тексту.

Предмет дослідження – метод ранжування тесту за їх відповідністю зображенню.

При досліджені задачі комп’ютерного зору та проаналізовано підходи до розпізнавання образів, а також методи обробки природних мов. За результатами аналізу показано важливість поєднання зображень та тексту, коли для кожного зображення задається речення, що описує це зображення.

РЕФЕРАТ

Пояснительная записка: 93 с., 27 рис., 2 прил., 31 источник.

АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ, ЗАДАЧА МАРКИРОВКИ ИЗОБРАЖЕНИЙ, КОГНИТИВНЫЕ СИСТЕМЫ, ОБНАРУЖЕНИЕ ОБЪЕКТОВ, ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА.

Цель данной квалификационной работы заключается в сочетании методов распознавания изображений и текста для того, чтобы промаркировать изображения предложениями, отражающие смысл этих изображений, что дает возможность построить когнитивный сервис, который имеет возможность одновременно работать и с изображениями и с текстом.

Объект исследования – комбинированный процесс распознавания изображений и текста.

Предмет исследования – метод ранжирования теста за их соответствием изображению.

При исследовании задачи компьютерного зрения и проанализированы подходы к распознаванию образов, а также методы обработки естественных языков. По результатам анализа показана важность совмещения изображений и текста, когда для каждого изображения задается предложения, описывает изображение.

ABSTRACT

Explanatory note: 93 pages, 27 figures, 2 annexes, 31 sources.

COGNITIVE SYSTEMS, IMAGE MARKING PROBLEM, NATURAL LANGUAGE PROCESSING, NEURAL NETWORK ARCHITECTURE, OBJECT DETECTION.

The aim of this certification work is to combine image and text recognition techniques in order to tag images with sentences that reflect the meaning of these images, which makes it possible to build a cognitive service that can simultaneously work with both images and text.

The object of study – combined image and text recognition process.

The subject of research – method of ranking tests for their relevance to the image.

In the study of the problem of computer vision, approaches to pattern recognition, as well as methods of processing natural languages, are analysed. Based on the analysis results, the importance of combining images and text is shown, when a sentence is specified for each image, an image describes.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

BRNN – Biside Recurrent Neural Network – двонаправлена рекурентна нейронна мережа;

CL – Computational Linguistics – обчислювальної лінгвістики;

CNNs – Convolutional Neural Networks – згорткові нейронні мережі;

MBL – Memory-Based Learning – навчання засноване на пам’яті;

MLP – Multilayer Perceptron – багатошаровий персептрон;

NLP – Natural Language Processing – обробка природних мов;

LSTM – Long-Short Term Memory – довга короткочасна пам'ять;

RNNs – Recurrent Neural Networks – рекурентні нейронні мережі;

SVM – Support Vector Machine – метод опорних векторів;

ВСТУП

Світова ІТ-індустрія наблизилася до вирішення задачі безпосередньої взаємодії з людиною на основі аналізу зображень, розпізнавання обличчя та емоцій, розпізнавання мови, аналізу та розуміння значення тексту, тощо. На сьогодні такі задачі вирішуються за допомогою когнітивних технологій та сервісів.

Термін «когнітивність» походить від латинського слова «cognoscere», що має значення «знати». Когнитивність зазвичай розглядається як здатність живих істот до обробки інформації, що надходить через органи чуття, набуття досвіду на базі цієї інформації та узагальнення отриманої інформації для оцінки, а також інтерпретації подій із навколишнього світу. Тому когнітивні сервіси пов'язані із набуттям та використанням знань, перетворенням всього обсягу інформації, який був отриманий із зовнішнього світу, у знання.

Когнітивні технології призначені для підтримки прийняття рішень у складній обстановці в реальному часі на основі алгоритмів, що працюють аналогічно людському мисленню. Використання когнітивних технологій направлено на оперативне виділення важливої інформації із наборів даних та подальше використання цієї інформації для вирішення прикладних задач.

Прогрес в апаратному забезпеченні, можливостях збору даних та алгоритмах їх обробки за останні роки значно приблизив вирішення задачі безпосередньої взаємодії систем штучного інтелекту з людиною. Найбільші досягнення на сьогодні отримані в області розпізнавання зображень. Існуючі алгоритми можуть класифікувати зображення на рівні людей-експертів, або з більш високою точністю. Вони використовуються в системах безпеки, електронної комерції, навіть у автомобілях з функцією автоматичного водіння. Ключова задача, яку вирішують ці методи – присвоєння міток зображенням. Кожна така мітка є назвою класу

зображення. Такий підхід відрізняється від процесу пізнання людини. Останній зв’язує візуальні дані і тексти або висловлювання природною мовою, забезпечуючи можливість переходу між областями зображень та їх текстового опису.

Таким чином, для поєднання візуальних об'єктів та тексту в задачах розпізнавання зображень необхідно або вирішити одну з двох задач: ідентифікувати зображення, які відповідають тексту, або виявити текст, який описує зображення.

Магістерська кваліфікаційна робота присвячена дослідженню задачі маркування існуючих візуальних об'єктів текстом. При вирішенні цієї задачі використовується модель, яка поєднує образи та текст. Тоді виникає можливість перейти від зображення до тексту, що визначає об'єкти на цьому зображенні. Такий перехід відбувається на основі ранжування. Зображення ранжується з урахуванням того, наскільки добре вони відповідають тексту. У магістерській кваліфікаційній роботі виконано удосконалення гібридної згорткової нейронної мережі, які дає можливість поєднати візуальні дані та тексти природною мовою.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Аналіз характеристик когнітивних систем та сервісів

Когнітивність представляє собою здатність розумної істоти до обробки та перетворення у знання інформації від органів органи чуття, а також використання цих знань для взаємодії із навколишнім світом, виділення важливих подій із навколишнього світу та їх подальшої інтерпретації. Когнітивні сервіси [1] забезпечують перетворення та використання знань, які були отримані із зовнішнього світу. Ці знання застосовуються для підтримки рішень людини у складній обстановці, що характеризується великою кількістю змінних, значення яких змінюються в реальному часі. Підтримка рішень виконується за допомогою алгоритмів, що функціонують аналогічно людському мисленню.

Пізнання як реальному, так і штучному світі реалізується, здебільшого, за допомогою побудови більш підходящих моделей та пошуку структур, які як найкраще описують ключові аспекти реальності. Нині, найбільш популярними та ефективними для вирішення такої задачі серед багатьох моделей штучного інтелекту (ШІ) є глибокі нейронні мережі [2], що рухають деякі з найбільш революційних технологій, що будуються в даний час. За останні роки в сфері когнітивних технологій набули також поширення статистичні моделі,та моделі засновані на правилах.

Побудова моделей є ключовим питанням не лише для когнітивних технологій, а й для науки та техніки в цілому. Хороші моделі мають наступні можливості:

– здатні представити відповідні характеристики складного явища в абстрактному представленні;

– дозволяють краще зрозуміти цільову систему;

– перевіряють прогнози, які можуть бути використані для підтвердження теорії.

Наприклад, маючи під рукою точну модель, можна управляти системою, прогнозуючи реакції на різні входи такої системи [3] і, таким чином, визначаючи набір входів, які можуть приймати систему, найближчу до бажаного стану.

Окрім широко використовуваних математичних та статистичних підходів у контексті когнітивних технологій, іншими прикладами є текстові описи та масштабні моделі. Однак одна особливість є спільною для всіх моделей: всі вони виділяють важливі аспекти системи, моделюється, спрощуючи та узагальнюючи властивості об'єктів реального світу. В результаті вважається, наприклад, що всі електричні батареї даного типу, вироблені на заводі, мають абсолютно однакову поведінку, таку ж як спостерігалась у наборі прототипів, які були протестовані та оцінені у лабораторних випробуваннях. Варто зауважити, що люди також схильні спрощувати реальність, наприклад, коли вважається, що користувачі можуть взаємодіяти з таким пристроєм, як смартфон, лише типовими способами, і ніхто не може використовувати його у новий, особливий спосіб. Коли розробляється типова поведінка користувача, вона узагальнюється й у результаті має відповідати послідовності дій ідеалізованого типового індивіду. Деякі з цих спрощень та невідповідностей вводяться через нестачу часу або ресурсів для проектування даних або знань, необхідних для створення більш точних моделей. Або ж причина може полягати у трудомісткості процесу розробки та створення моделі, що охоплюють більше аспектів системи, що повинна бути представлена.

До недавнього часу для нової конструкції системи потрібна була лише теорія, яка б приблизно пояснювала її поведінку та спосіб її

використання, однак такого підходу вже недостатньо. Через широке впровадження технологій збільшується непередбачуваність їх використання, але дедалі швидший темп технологічних та соціальних змін робить моделі легко застарілими [4]. Також, зростає потреба у більшій точності та ефективності роботи систем, у розробці програм, що вирішують більш складні завдання, у яких людям бракує повного розуміння. Наприклад, обробка мови та зображень, вимагають від інженерів пошуку нових типів моделей.

У такому випадку, для вирішення розглянутих проблем використовуються когнітивні технології, в процесі розробки яких досліджуються вимоги, варіанти проектування та компроміси для людських, автономних, інтегрованих, фізичних систем, включаючи вимоги до архітектур [5], для форм представлення, для механізмів сприйняття, для навчання, планування, міркування, дій та спілкування. Тобто, необхідна розробка достатньо вимогливих методів та підходів, за допомогою яких машина не тільки виконує певне завдання, але й показує розуміє розуміння, що вона зробила, і чому.

Враховуючи розглянуті вимоги, найбільш ефективним є використання когнітивних систем, що пов’язані з машинним навчанням. Такі системи відносяться до адаптивності як до найважливішої риси для розробки кращих систем, зменшуючи їх залежність від знань, наявних під час проектування. Когнітивний підхід передбачає, що, якщо не можливо апріорі визначити, як поводитиметься система та навколишнє середовище, якій вона буде відповідати, такі системи повинні мати можливість збирати дані та автономно адаптуватися до спостережуваної реальності, передбачаючи майбутні стани, коли це можливо.

У свою чергу, незважаючи на вражаючий прогрес у багатьох конкретних підтемах у галузі штучного інтелекту та когнітивної науки, сфера в цілому рухається повільно. Частково причиною цього є те, що за

останні кілька десятиліть дослідження стали фрагментованими: багато людей та дослідницькі групи зосереджували свої зусилля на вузько визначених проблемах наприклад комп’ютерного зору, або навчання, або розумінні мови, або вирішенні проблем.

Ключовою метою когнітивних систем було подолати деякі з цих обмежень, використовуючи ідеї відповідних дисциплін, щоб дослідити амбіційне бачення висококваліфікованої системи, яка поєднує в собі набір різних можливостей, наприклад, підмножину можливостей типової людини 4 – 5-річної дитини, з точки зору когнітивних функцій мозку. Наукова важливість цієї мети полягає в тому, що така система вимагає загальних можливостей, що забезпечують платформу для багатьох різних видів подальшого розвитку, оскільки дитина цього віку може розвиватися в будь-якій людській культурі та отримувати користь від багатьох форм навчання. Однак зазвичай подібна задача розглядається поверхнево, не вдаючись до певних й дуже важливих деталей, які, як правило недооцінюються. Подібні дослідження використовують результати та ресурси різних дисциплін, проте найважливішу роль грають саме штучний інтелект та когнітивна наука, наприклад, нові результати щодо сприйняття, навчання, міркування, обробки мови, пам'яті, виконання плану та дослідження мотивації та емоцій. Тобто, подібні проекти не лише отримують користь від інших дисциплін, але також вносять свій вклад у їх розвинення, намагаючись надати нові основні внески до цих дисциплін у вигляді нових теорій та робочих моделей. Очікується, що результати будуть корисні як для вдосконалення наукового розуміння природних інтелектуальних систем (наприклад, людей та інших тварин), так і для проектування штучних інтелектуальних систем.

Прикладом когнітивного сервісу є система IBM Watson. Вона орієнтована на пошуки відповідей на запитання користувачів та фактично пройшла тест Тьюринга в 2011 році, коли виграла в людей у гру Jeopardy.

В цій грі система відповідала на питання з різний областей знань. Система переводила голос в текст та «розуміла» сенс цього тексту, використовуючи закладені в неї знання. Вона вибирала найбільш правильну відповідь із можливих варіантів, пропонуючи цю відповідь людині. На сьогодні когнітивні сервіси від цієї системи використовуються в медицині при розробці персоніфікованого курсу лікування з урахуванням історії хвороби, а також характеристик пацієнтів. Також даний сервіс використовується в Австралії у компанії Woodside Energy для пошуку корисних копалин. Використання системи дозволило зменшити витрати на підготовчу роботу до пошуку з 80 до 20% від вартості проекту. Даний когнітивний сервіс використовується також у банківській сфері, в ритейлі, в управлінні енергетикою. Зокрема, IBM впровадила автоматизацію контрактів у сфері IT з використанням блокчейн-технології у Bank of Tokio-Mitsubishi. Також когнітивними сервісами користується Німецька біржа. Інструменти побудови когнітивних сервісів пропонує також платформа Microsoft Azure. Використовуються сервіс Azure Bot та спеціалізована платформа Bot Framework. Ці засоби призначені для побудови корпоративних ботів з підтримкою взаємодії за допомогою природної мови. Когнітивні сервіси забезпечують аналіз запитів. При використанні цих сервісів бот дає «розумні» відповіді на запитання.

За оцінками Microsoft, зараз до 75% додатків цієї фірми використовують когнітивні сервіси у фоновому режимі, тобто користувач у більшості випадків на знає про це. Когнітивні сервіси Microsoft вирішують такі задачі:

– підтримка прийняття рішень, в тому числі виявлення аномалій;

– вилучення сенсу із тексту з подальшими візуальними або звуковими підказками читачеві;

– інтеграція обробки природної мови в існуючі сервіси та додатки;

– аналіз візуального контенту на основі розпізнавання зображень;

– інтелектуальний пошук по зображенням, записам звуку, з урахуванням контексту.

В цілому когнітивні сервіси використовують ідею самомодифікуючої архітектури, що включає різні типи можливостей, які розвиваються з часом [5], [6].

В основі таких проектів лежать методи з наступними властивостями:

– забезпечують автоматичну інтерпретацію навколишнього середовища (розпізнавання великої кількості об'єктів тощо);

– забезпечують адаптивне набуття нових навичок та завдань у співпраці з користувачем-людиною;

– вдосконалюють методи управління, що дозволяє системі виконувати більш складні задачі від користувачів;

– мають можливість вдосконалення методів логічного виводу для забезпечення розширеної автономності.

Інтерпретаційні засоби можна використовувати:

– для забезпечення автономності;

– для передачі знань та параметрів задачі користувачеві.

У свою чергу, для того, щоб система мала змогу правильно функціонувати та не шкодила оточуючим при виконанні своєї функції, важливо, щоб вона мала засоби для автоматичної адаптації до середовища. Також повинна мати можливість адаптуватися до звичок власника, вона повинна мати можливість розуміти важливість та наслідки інструкцій, використовуючи достатньо гнучкий підхід для засвоєння нових навичок.

Загальна мета такої системи розбивається на під задач [7] для більш легкої і зрозумілої реалізації. Таким чином, аналізуючи численні вимоги до задачі в цьому напрямку, можна отримати послідовно менш складні під цілі, які забезпечують кроки до віддаленої мети. Деякі з цих під цілей є досяжними за певний час, й одна за таких під цілей, на сьогоднішній день є досить популярною задачею комп’ютерного зору, а саме розпізнавання

зображень, для вирішення якої існує досить багато підходів, проте, також існує багато недоліків, які можна вдосконалити.

1.2 Дослідження задач комп’ютерного зору

Однією з цілей галузі штучного інтелекту є надання комп’ютерам можливості побачити і зрозуміти світ навколо нас з візуальної точки зору [8], [9] і наділити їх здатністю спілкуватися з нами природною мовою. Людям легко виконувати найрізноманітніші завдання, що включають складне розпізнавання зору та розуміння оточення, завдання, що передбачають спілкування природною мовою, та завдання, що поєднують переклад між двома способами. Наприклад, швидкого погляду на зображення достатньо, щоб людина вказала і описала величезну кількість деталей про оточення. Проте настільки ж природний для людини цей процес, настільки ж він є складним для комп’ютерів.

Комп’ютерний зір – це автоматичний аналіз зображень та відео за допомогою комп’ютерів, з метою отримання певного розуміння світу. Комп’ютерний базується на можливостях системи людського зору. Коли його почали розробляти в 1960-х та 1970-х роках, зір вважали відносно простою проблемою для вирішення. Однак причина цьому полягала у наявності нашої власної зорової системи, яка робить завдання інтуїтивним для нашого свідомого розуму. Проте, насправді зорова система людини дуже складна, і навіть оцінки того, скільки мозку задіяно в зоровій обробці, коливаються від 25% до понад 50%.

Коли комп'ютерний зір вперше розпочався на початку 1970-х, він розглядався як компонент візуального сприйняття оточуючого середовища для імітації людського інтелекту та наділення роботів розумною поведінкою. У той час деякі науковці того часу у сферах штучного інтелекту та робототехніки, в таких університетах, як MIT та Stanford

вважали, що вирішення проблеми «візуального введення» буде простим кроком на шляху до вирішення більш складних такі проблеми, як міркування та планування вищого рівня [10]. На рисунку 1.1 зображена шкала часу з найбільш значущими дослідженнями у сфері штучного інтелекту у період з 1970 по 2000-ні.

Рисунок 1.1 – Шкала часу з найбільш значущими дослідженнями у сфері компьютерного зору

Саме бажання створити примітивну структуру світу із образів, і використовувати це як проміжну мету до повного розуміння світу, відрізняло комп’ютерне бачення від уже існуючої галузі цифрової обробки зображень.

Ранні спроби розуміння передбачали вилучення країв, а у результаті цього створювати трьохмірну структуру об’єктів з топологічної структури двохмірних ліній. У той час було розроблено кілька алгоритмів маркування ліній, приклад маркування ліній довільної форми зображена на рисунку 1.2.

Також вивчалось тривимірне моделювання небагатогранних об'єктів [8], [9], [11]. Один з популярних підходів використовував узагальнені циліндри, тобто тверді тіла обертання і замічені замкнуті криві, часто розташовані по частинах.

Вони також називалися зображувальними структурами, приклад зображений на рисунку 1.3. Такий підхід використовує процес декомпозиції складної фігури на більш прості.

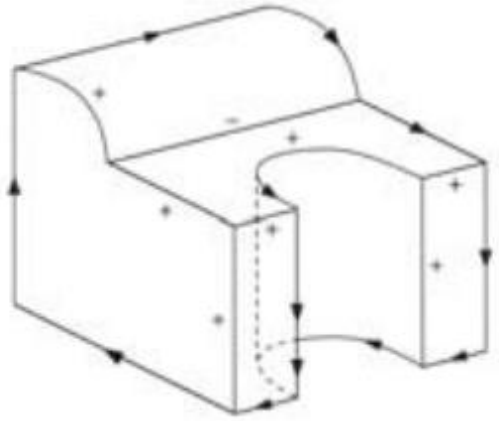

Рисунок 1.2 – Маркування ліній довільної форми

Варто зауважити, що в даний час це один із вживаних підходів, який використовується при розпізнаванні об'єктів.

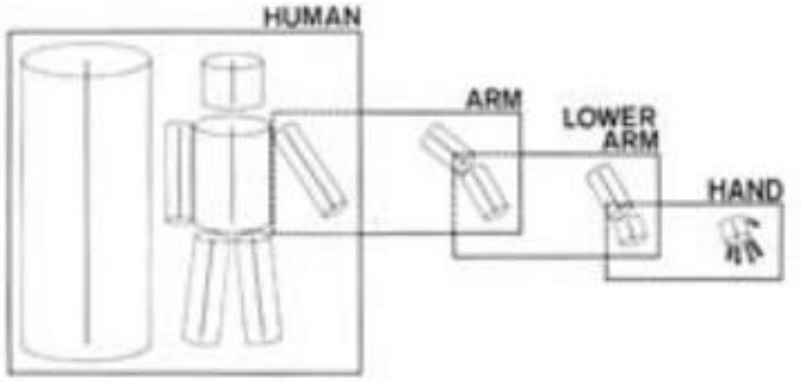

Рисунок 1.3 – Моделювання за допомогою зображувальних структур

Також розглядалися підходи, які б пояснювали інтенсивність та варіації затінення – рисунок 1.4.

Рисунок 1.4 – Приклад варіацій інтенсивності затінення

У той час були також розроблені такі підходи [10], [11], як:

– перший із багатьох функціональних стерео алгоритмів відповідності;

– алгоритми оптичного потоку на основі інтенсивності – рисунок 1.5;

– ранні роботи з одночасного відновлення тривимірної структури та руху камери.

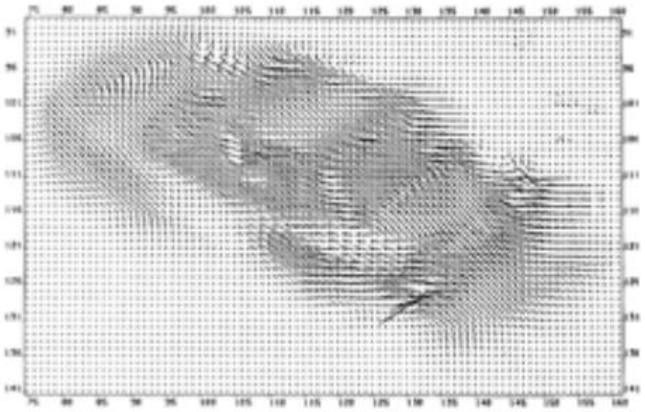

Рисунок 1.5 – Приклад роботи алгоритму оптичного потоку на основі інтенсивності

У цей час з’явилось дуже багато робіт присвячених дослідженню механізму зору як складової розуміння людини. Зокрема, відомий нейробіолог Девід Марр представив своє уявлення про три рівні опису (візуальної) системи обробки інформації:

– обчислювальна теорія, мета якої обчислення і визначення обмежень, що можуть бути застосовані до вирішуваної проблеми;

– представлення та алгоритми, тобто представлення вхідної, вихідної та проміжної інформації та які алгоритми використовуються для отримання бажаного результату;

– апаратна реалізація – спосіб відображення та як алгоритми працюють на обладнанні.

У 1980-х роках головну увагу було зосереджено на більш досконалих математичних методах для кількісного аналізу зображень та сцен. Піраміди зображень почали широко використовуватися для виконання таких завдань, як змішування зображень та пошук, від грубого до тонкого пошуку відповідності.

Також були розроблені безперервні версії пірамід, що використовують концепцію обробки масштабного простору. Наприкінці 1980-х років використання стерео сигналу як кількісного сигналу фігури було розширено за допомогою широкого розмаїття технік фігури-Х, включаючи фігуру від затінення що зображена на рисунку 1.6, форма з текстури і форма з фокусу. У цей період також проводились дослідження щодо кращого виявлення країв та контурів.

У цей час, багато алгоритмів стерео, потоку та виявлення ребер були уніфіковані, також було зазначено, що такі проблеми можна однаково добре сформулювати, використовуючи дискретні моделі випадкового поля Маркова, що дозволило використовувати кращий (глобальний) пошук та оптимізаційні алгоритми.

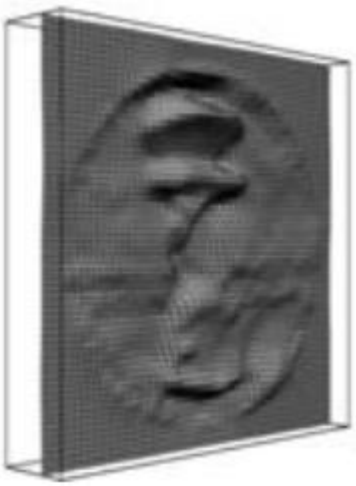

Рисунок 1.6 – Приклад фігури затінення

У 1990-ті був сплеск активності з використанням проективних інваріантів для розпізнавання перетворився на спільні зусилля для вирішення структури з проблеми руху.

Робота, розпочата у 1980-х роках над використанням детальних вимірювань кольору та інтенсивності, у поєднанні з точними фізичними моделями переносу сяйва та формування кольорового зображення створила власне підполе, відоме як бачення на основі фізики.

Також значно покращились алгоритми відстеження, включаючи відстеження контурів за допомогою активних контурів таких як, фільтри частинок та набори рівнів, а також прийоми, що базуються на інтенсивності, часто застосовуються для виявлення облич приклад зображений на рисунку 1.7.

Мабуть, найбільш помітним розвитком комп'ютерного зору протягом цього десятиліття стала посилена взаємодія з комп'ютерною графікою [12], особливо в міждисциплінарній галузі моделювання на основі зображень. Ідея маніпулювати реальними образами безпосередньо для створення нових анімацій вперше виникла за допомогою технологій морфінгу

зображень, а пізніше була застосована для інтерполяції, зшивання панорамних зображень, а також візуалізація в повному світлі.

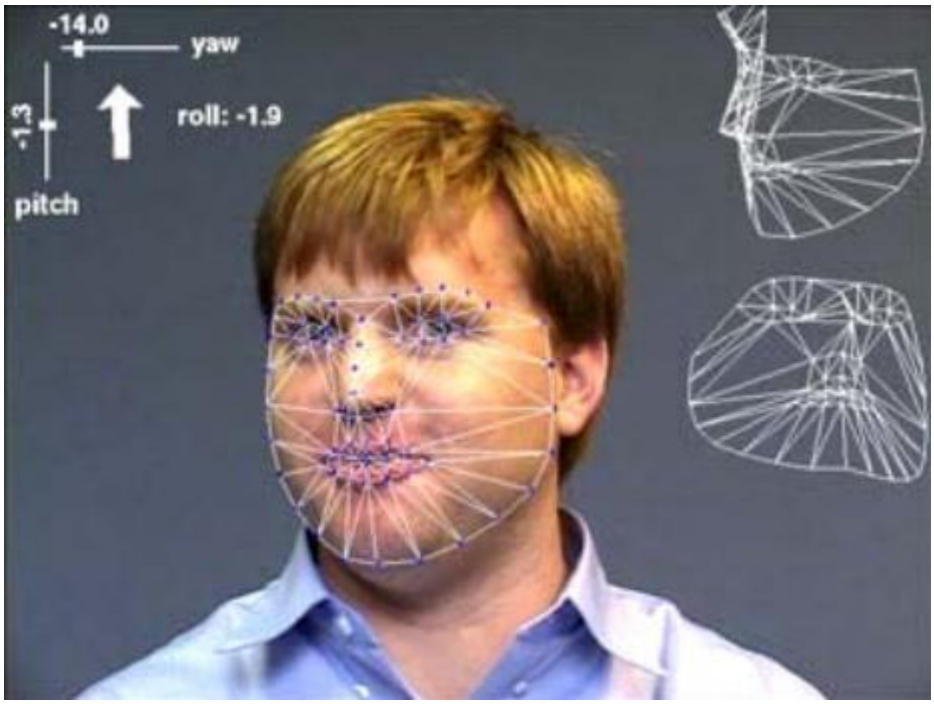

Рисунок 1.7 – Приклад виявлення обличчя за допомогою активних контурів

У 2000-ні спостерігається поглиблення взаємодії між полями зору та графіки. Зокрема, багато тем, представлено під рубрикою візуалізації на основі зображень, таких як зшивання зображень, захоплення та рендеринг світлового поля та захоплення зображення з високим динамічним діапазоном, об'єднання декількох експозицій [13]. Були розроблені методи об'єднання спалахів із не блискавими аналогами. Синтез текстур та фарбування – це додаткові теми, які можна класифікувати як обчислювальні техніки фотографії, оскільки вони повторно поєднують зразки вхідних зображень для створення нових фотографій.

Ще однією суттєвою тенденцією минулого десятиліття стала розробка більш ефективних алгоритмів для складних глобальних задач оптимізації, збільшення інтересу у розпізнаванні образів застосовуючи

складні методи машинного навчання [11], [14] до проблем комп'ютерного зору.

Другою помітною тенденцією стала поява на основі ознак методів розпізнавання об'єктів, сцен, панорам та розташування. В останній час дослідження сфокусовані на сегментації регіону, приклад якого зображено на рисунку 1.8, проте початок був покладений саме у цей період.

Рисунок 1.8 – Приклад сегментації регіону

На сьогоднішній день комп’ютерний зір використовується в широкому діапазоні реальних задач, які включають:

– розпізнавання символів: читання рукописних поштових індексів на буквах та автоматичне розпізнавання номерних знаків;

– перевірка: швидка перевірка деталей для забезпечення якості, наприклад, для вимірювання допусків на крилах літаків або частинах кузова автомобіля, або пошуку дефектів у сталевих виливках;

– роздрібна торгівля: розпізнавання об’єктів для автоматизованих смуг сортування;

– побудова тривимірних моделей: повністю автоматизована побудова тривимірних моделей з аерофотознімків, що використовуються в онлайн мапах;

– медична візуалізація: реєстрація доопераційних та зображень, що робляться під час операції або проведення довгострокових досліджень стану мозку людей по мірі змінення стану пацієнта;

– автомобільна безпека: виявлення несподіваних перешкод, таких як пішоходи на вулиці, в умовах, коли методи активного зору, такі як радар або лідар, не працюють належним чином ;

– об’єднання комп’ютерних зображень із кадрами в реальному часі шляхом відстеження точок об’єктів у вихідному відео для оцінки руху 3D-камери та форми навколишнього середовища;

– зйомка руху: використання світло відбивних маркерів, що переглядаються з декількох камер, або інших методів, що базуються на зорі, для захоплення акторів для комп'ютерної анімації;

– моніторинг зловмисників, аналіз дорожнього руху та моніторинг басейнів;

– розпізнавання відбитків пальців та біометрія: для автоматичної аутентифікації доступу, а також для судових програм.

Таким чином, сфери використання комп’ютерного зору охоплюють широкий спектр напрямків людської діяльності.

1.3 Аналіз підходів до розпізнавання образів

Зображення – це підмножина сигналів, що є функцією, яка передає інформацію загалом про поведінку фізичної системи або атрибути якогось явища. Наприклад, сигнал дорожнього руху, який використовує три універсальні кольорові коди – червоний, жовтий та зелений, що сигналізують про момент зупинки, їзди або прогулянки. Хоча сигнали

можуть бути представлені різними способами, у всіх випадках інформація міститься у варіації шаблонів, які приймають різні форми, передаючи та приймаючи інформацію.

Тобто, приймаючи до уваги той факт, що зображення-сигнали можуть бути представлені у різних формах, вони можуть мати різні розміри, якість, можуть бути забарвлені, а можуть бути чорно-білі; зображення можуть надходити один за одним у потоці з великою швидкістю, приймаючи форму відео, це все ускладнює вирішення задачі розпізнавання. В цілому задача розпізнавання зображень має більш загальний характер [15], [16], [17]. Проте якщо розглядати навіть підзадачі, наприклад знаходження певних патернів у відео потоці, тоді розпізнавання зображень не здається настільки легкою. Крім того, якщо розглянути певну задачу більш детально, то з’являться все більше деталей та проблем, на які необхідно звернути увагу.

Для розпізнавання зображень потрібні алгоритми, які одночасно надійні проти шуму та відхилення від наших моделей, а також досить ефективні з точки зору ресурсів [18]. Зокрема, найсучасніші моделі розпізнавання зображень, засновані на глибоких згорткових нейронних мережах, стали здатними виділяти тисячі візуальних категорій з точністю, порівнянною з людьми, або навіть перевершувати їх у деяких дрібнозернистих категоріях, таких як породи собак [19].

Однак домінуючим підходом у більшості методів є моделювання проблеми розпізнавання зору як завдання класифікації зображень на деяку кількість фіксованих та жорстко закодованих візуальних категорій. Наприклад, якщо на передньому плані зображений малюк, який грає на фоні будинку, що оточений деревами, то у такому випадку система буде класифікувати таке зображення як «малюк», замість більш детального – «малюк», «іграшки», «будинок», «дерево». Тобто, такий підхід менш

складний й не може повноцінно вирішувати задачу порівняно з більш комплексним методом, який називається Dense Captioning.

З метою вирішення задачі розпізнавання зображень, а саме використовуючи підхід Dense Captioning були обрані нейронні мережі, які забезпечують високу якість, що вирішуючи цю задачу у певних сферах є пріоритетом, наприклад, у машинах без водія. Отже, розглядаються наступні нейронні мережі:

– Vanilla Neural Network, яка більш відома під назвою «Багатошаровий персептрон»;

– згорткові нейронні мережі (Convolutional Neural Networks);

– рекурентні нейронні мережі (Recurrent Neural Networks).

Багатошаровий персептрон (MLP, або Vanilla Neural Network) – це клас штучної нейронної мережі, а саме прямого поширення. Багатошаровий персептрон зображено на рисунку 1.9.

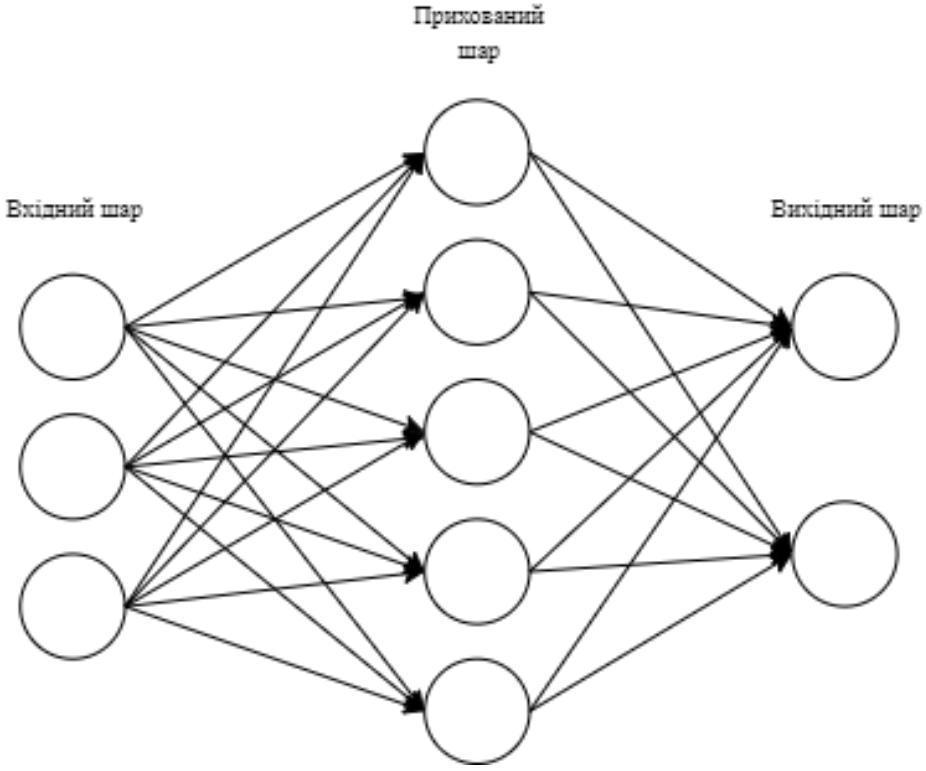

Рисунок 1.9 – Загальний вигляд архітектури багатошарового персептрону

Його схема складається щонайменше з трьох шарів: вхідного шару, прихованого шару та вихідного шару. За винятком вхідних вузлів, кожен вузол є нейроном, який використовує нелінійну функцію активації. MLP використовує контрольовану техніку навчання, вона також називається навчання з вчителем, яка називається зворотним розповсюдженням для навчання. Його багатошаровість і нелінійна активаційна функція відрізняють MLP від лінійного персептрона. Він може розрізняти дані, які не можна лінійно розділити, наприклад задача XOR не може бути вирішена за допомогою лінійного перстептрону, адже на двомірній площині неможливо лінійно розділити в рамках поставленої задачі, проте якщо розглянути у трьохмірній площині та використати нелінійну функцію активації, то ця задача може бути вирішена.

Нейронна мережа Vanilla має повністю з'єднані шари, де кожен персептрон пов'язаний з кожним іншим персептроном. Недоліком є те, що кількість загальних параметрів може зрости до дуже великої (кількість персептронів в першому шарі помножена на кількість в другому шарі, помножена на кількість персептронів в третьому шарі). Подібна архітектура неефективна, оскільки система може страждати від прокляття розмірності, якщо вона буде працювати з реальними задачами. Також, ще одним недоліком є те, що багатошаровий персептрон нехтує просторовою інформацією, приймаючи сплощені вектори як вхідні дані.

Не роблячи будь-яких припущень про те, в якій формі будуть представлені вхідні дані, тобто про структуру *x*, будується нейронна мережа, яка повторює множення матриць та нелінійність кожного елемента. Наприклад, двошарова нейронна мережа може бути реалізована наступним чином:

$$f(x) = W_2\sigma(W_1 x), \tag{1.1}$$

де W_1, W_2 – матриці,

σ – це нелінійність елемента (наприклад, tanh).

У свою чергу, тришарова мережа мала б вигляд

$$f(x) = W_3\left(W_2\sigma(W_1 x)\right). \tag{1.2}$$

Також варто зауважити, що у випадку, якщо нелінійність є функцією ідентичності, то вся нейронна мережа зменшується до лінійного представлення. Функціями активації є тангенс гіперболічний, сигмоїдна функція та ReLU. Також, останній шар нейронної мережі зазвичай не містить нелінійності, а одношарова нейронна мережа – це просте лінійне перетворення.

Багатошаровий персептрон використовується для застосування в комп'ютерному зорі. Проте самої цієї системи недостатньо для вирішення здачі розпізнавання, тому його наслідує згорткова нейронна мережа (CNN).

Згорткові нейронні мережі (CNN або Convolutional neural networks) – нейромережева система й на сьогоднішній найбільш популярний алгоритм серед існуючих у сфері комп'ютерного зору. Кожен фільтр CNN переміщується навколо всього зображення відповідно до певного розміру та кроку, дозволяє фільтру знаходити та узгоджувати шаблони незалежно від того, де шаблон знаходиться на даному зображенні. Шари зв'язані, проте вони не повністю, частково з'єднані [15], на противагу багатошарового персептрону; кожен вузол не підключається до будь-якого іншого вузла.

Панорамування фільтрів (ви можете встановити крок і розмір фільтра) в CNN, по суті, дозволяє розподіляти параметри, розподіляти вагу, щоб фільтр шукав певний шаблон і міняв місце розташування – може знаходити шаблон де завгодно на зображенні. Це дуже корисно для

виявлення об'єктів. Візерунки можна виявити в більш ніж одній частині зображення.

Згорткові нейронні мережі спеціально розроблені для обробки даних, які мають деяку просторову топологію, наприклад, зображень, відео, спектрограм звуку при обробці мови, або послідовностей символів у тексті. У кожному з цих випадків вхідним прикладом x є багатовимірний масив, (який також має назву «тензор»). Наприклад кольорове зображення, яке має розмірність 256×256 пікселів – це тензор, адже через те, що зображення кольорове, воно має три канали RGB, тобто червоний, зелений та голубий. У такому випадку розмірність зображення помножається на три: 256 × 256 × 3.

У випадку із звуковою спектрограмою може бути масив розміром 1 000 × 128, що вказує амплітуду будь-якої з 128 частот у будь-який момент часу від t = 1 до t = 1000. Деякі речення можуть бути представлені на рівні символів як 112 × 30, вказуючи, який із 30 можливих символів займає будь-яку з 112 позицій у реченні. У багатьох із цих випадків вхідна розмірність висока, наприклад, зображення вище матиме приблизно 200 000 чисел. Тому недоцільно, як за кількістю параметрів, так і за часом обробки, використовувати повнозв'язані шари [14]. У таких випадках віддається перевага розробці архітектур нейронних мереж, які знають про просторову схему введення та використовують конкретні локальні можливості зв’язку та розумні схеми спільного використання параметрів.

Проста згорткова нейронна мережа – це послідовність шарів, і кожен шар цієї системи перетворює один обсяг активацій в інший через диференційовану функцію. Використовується три основні типи шарів для побудови архітектури нейронної мережі: згортковий шар (Convolutional layer), шар пудлінгу (pooling) та шар до якого підключені всі сигнали з попереднього шару (Fully-Connected Layer), такий шар присутній у всіх звичайних нейронних мережах.

Наприклад, проста класифікуючи згорткова нейронна мережа може мати архітектуру, що представлена на рисунку 1.10.

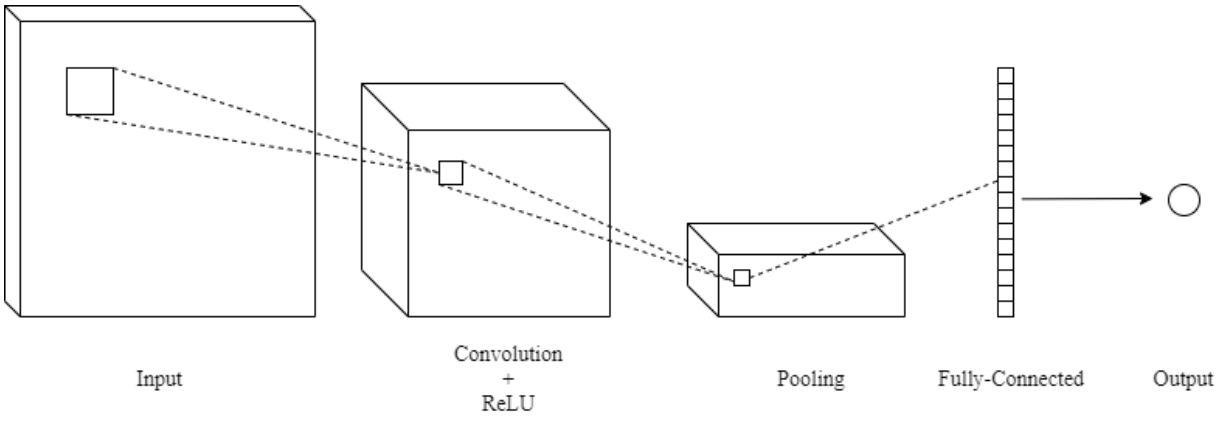

Рисунок 1.10 – Загальний вигляд архітектури багатошарового персептрону

Вхід системи може приймати вихідні значення пікселів зображення, в даному випадку зображення шириною 32, висотою 32 і з трьома кольоровими каналами R, G, B.

Рівень згортки обчислює вихід нейронів, які підключені до локальних областей на вході, кожен обчислює точковий добуток між їх вагами та малою областю, з якою вони підключені у вхідному обсязі, що може призвести до об’єму, такого як (32×32×12), якщо використовувалося 12 фільтрів. Далі на рівні ReLU застосовується елементна функцію активації, а саме порогове значення *max (0, x)* при нулі, що залишає розмір незмінним – (32×32×12).

Шар пулінгу виконує операцію зменшення дискретизації вздовж просторових розмірів (ширина, висота), що призводить до об'єму, такого як (16×16×12).

Рівень FC обчислює оцінки класу, що призводить до об’єму розміру (1×1×10), де кожне з 10 чисел відповідає оцінці класу, як і у звичайних

нейронних мережах, і як випливає з назви, кожен нейрон у цьому шарі буде підключений до всіх чисел у попередньому.

Таким чином, згорткові нейронні мережі перетворюють шар за шаром із вихідних значень пікселів у кінцеві оцінки класу. У свою чергу, деякі шари містять параметри, а інші – ні. Зокрема, рівні конволюції та FC виконують перетворення, які є функцією не тільки активацій у вхідному обсязі, але й ваг. З іншого боку, шари ReLU та пулінгу реалізовують фіксовану функцію, а параметри в шарах конволюції та FC будуть тренуватися з градієнтним спуском, щоб оцінки класу, які обчислює мережа, відповідали міткам у навчальному наборі для кожного зображення.

1.4 Методи обробки природних мов

Область обчислювальної лінгвістики (computational linguistics, або CL) разом з її інженерною областю обробки природних мов (natural language processing, NLP) за останні роки стали дуже популярними. Область обчислювальної лінгвістики стала не тільки самостійною науковою дисципліною, а також важливим напрямком промислового розвитку. За останні три десятиліття фокус [15] досліджень в CL та NLP перемістився від вивчення малих прототипів та теоретичних моделей до надійних систем навчання та обробки, що застосовуються у великих корпораціях.

Теорії формальної мови є однією з основних формальних опор обчислювальної лінгвістики. Ізраїльський науковець Шулі Вінтнер пропонує надзвичайно чітке керівництво по класичних мовних класах ієрархії Хомських, і він демонструє взаємозв'язок між цими класами граматиками, що розпізнають їх членів.

Хоча формальна теорія мови ідентифікує класи мов та їх прийнятність (або відсутність таких), теорія складності вивчає обчислювальні ресурси в часі та просторі, необхідні для обчислення елементів цих класів. Ян Пратт-Хартманн вводить цю центральну область інформатики, і бере на себе її значення для CL та NLP. Він описує ряд важливих результатів складності для кількох видатних класів мови та завдань NLP. Він також поширює трактування складності [16] в CL / NLP від класичних проблем, таких як синтаксичний розбір, до відносно невивченої області обчислення значення речення та логічних зв'язків між реченнями.

Проте одним з найважливіших досліджень є робота Марка-Яном Недергофом та Джорджо Саттою про формальні основи синтаксичного аналізу, які ілюструють проблему розпізнавання та репрезентації синтаксичної структури з вивченням безконтекстних граматик та табличних розборів. Вони представляють кілька алгоритмів синтаксичного аналізу безконтекстних граматик, і вони розглядають імовірнісний синтаксичний аналіз безконтекстних граматик. Потім вони поширюють своє дослідження на синтаксичні аналізатори граматики залежностей та граматика складання дерева.

Розвиваючись, дана наука прийшла до моделей максимальної ентропії, які становлять важливу техніку машинного навчання, що передбачає мінімізацію упередженості в моделі ймовірності для набору подій до мінімального набору обмежень, необхідних для розміщення даних. У процесі розвитку цієї ідеї була порівняна MaxEnt з SVM (support vector machine) – іншої техніки ML, де була розглянута корисність у частині тегів мови, синтаксичного аналізу та машинного перекладу.

Наступними з’явилися моделі класифікації ML з навчанням на основі пам'яті (memory-based learning, MBL), які широко використовується в обробці природніх мов. Memory-based learning використовує міру

схожості, щоб оцінити відстань між векторами ознак, що зберігаються в навчальних даних, щоб побудувати класифікаційні класи [17]. Така система навчання є дуже універсальною та ефективною альтернативою методам статистичного моделювання мови.

Наступним кроком був розглядання модифікованої та розширеної версії memory-based learning, а саме фонологічний та морфологічний аналіз, тегування частини мови, неглибокий синтаксичний розбір, неоднозначність слів, фрагментація фрагментів, розпізнавання іменних сутностей, генерація, машинний переклад та розпізнавання діалогових актів, які можуть бути застосовані до широкого кола задач.

Також були розглянуті прості дерева прийняття рішень для вирішення проблеми аналізу мови, проте вони часто виявляють нестабільність через їх чутливість до невеликих змін у характеристиках даних. Тому були розглянути модифікації дерев рішень, які долають це обмеження, методи, що поєднують набори дерев, індуковані для набору даних, для досягнення більш надійного класифікатора.

Наступним кроком еволюції методів NLP були штучні нейронні мережі, зокрема в рамках цієї проблеми були розглянуті багатошарові персептрони (multi-layered perceptrons, MLP), які містять приховані шари між входами та виходами, та реккурентні MLP, які мають циклічні зв'язки з прихованими одиницями. Такі зв’язки дозволяють системі обробляти необмежені послідовності, зберігаючи копії прихованих станів одиниць і подаючи їх назад як вхідні дані для одиниць, коли вони обробляють послідовні позиції в послідовності [18], [19]. Фактично вони забезпечують систему пам'яттю для обробки послідовностей входів. Було проілюстровано застосування нейронних мереж до завдань генерації статистичних мовних моделей для набору даних, вивчення різних видів синтаксичного синтаксичного аналізу та визначення семантичних ролей.

Звичайна нейронна мережа, варіації архітектури якої представлені на рисунку 1.11 та рисунку 1.12, може взяти вхідний вектор, перетворити його через певний прихований шар і створити вихідний вектор. На вхід стандартної нейронної мережі надходить лиш один вектор.

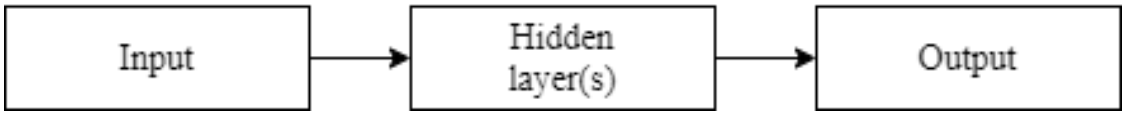

Рисунок 1.11 – Загальний вигляд стандартної нейронної мережі

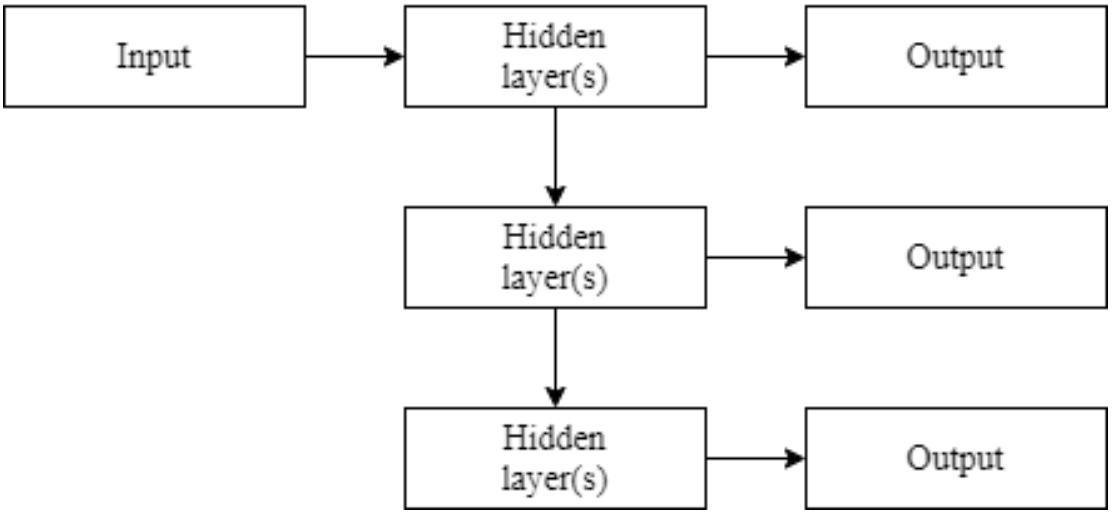

Рисунок 1.12 – Багатошарове представлення стандартної нейронної мережі

Проте на діаграмах, що приведені на рисунку 1.13, рисунку 1.14 та рисунку 1.15 рекурентні нейронні мережі дозволяють обробляти послідовності векторів. Наприклад, відповідно до рисунку 1.13 на виході, відповідно до рисунку 1.14 на вході або відповідно до рисунку 1.15 послідовно, так і паралельно.

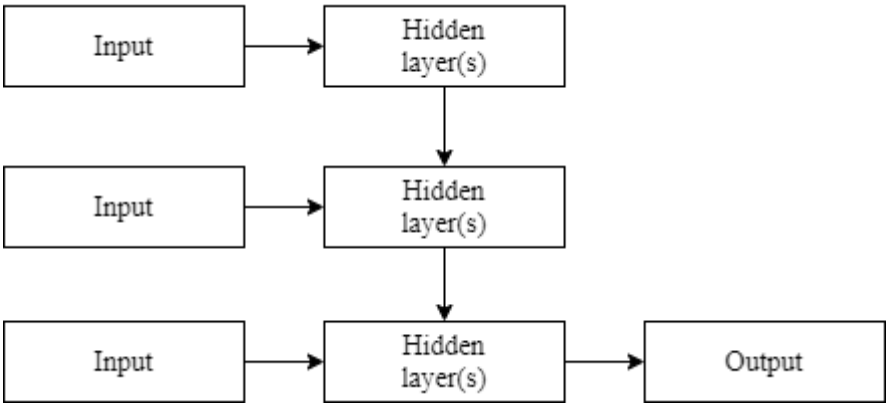

Рисунок 1.13– Обробка послідовності векторів на виході системи за допомогою RNN

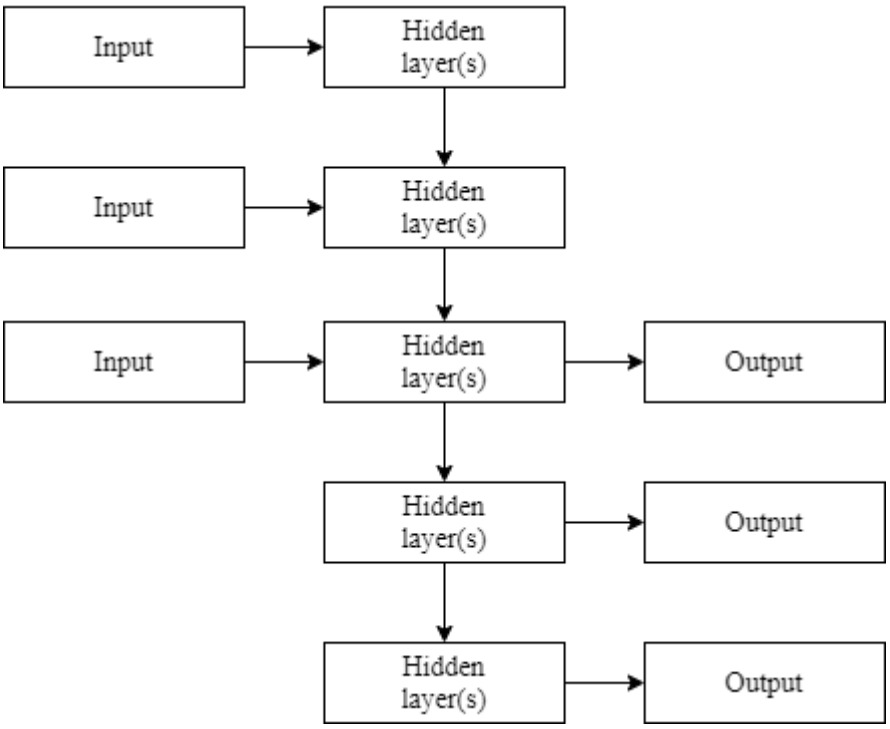

Рисунок 1.14– Обробка послідовності векторів на вході системи за допомогою RNN

Такому типу обробки сприяє повторюваний прихований шар.

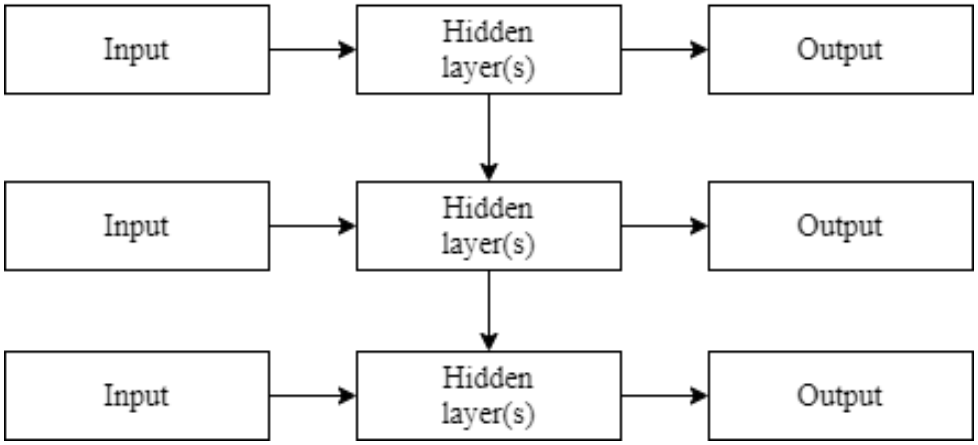

Рисунок 1.15 – Паралельна обробка послідовності векторів за допомогою RNN

Даний шар маніпулює набором внутрішніх змінних h_t на основі попереднього прихованого стану h_{t-1} та поточного вводу, використовуючи фіксовану формулу реккурентності:

$$h_t = f_\theta(h_{t-1}, x_t) \tag{1.3}$$

де θ – параметри, які використовуються на кожному часовому кроці, що дозволяє обробляти послідовності з довільною кількістю векторів.

Отже, рекурентна нейронна мережа (RNN) [16], [20] – це модель зв'язку, яка обробляє послідовність векторів $\{x_1, \ldots, x_T\}$ з використанням формули рекурентності.

Прихований вектор h_t можна інтерпретувати як поточний підсумок усіх векторів x, поки цей крок часу і формула повторення не оновлюють зведення на основі наступного вектора. Зазвичай використовують або $h_0 = \vec{0}$, або трактують h_0 як параметри і вивчають початковий прихований стан. Точна математична форма рекурентності змінюється в залежності від моделі.

Ванільна рекурентна нейронна мережа (Vanilla RNN) використовує рекурентість у формі:

$$h_t = \tanh\left(W\left(x_t, h_{t-1}\right)^T\right) \tag{1.4}$$

Тобто попередній прихований вектор і поточний вхід об'єднуються і трансформуються лінійно за параметрами *W*, що еквівалентно замість:

$$h_t = \tanh\left(W_{xh}x_t + W_{hh}h_{t-1}\right) \tag{1.5}$$

де дві матриці W_{xh}, W_{hh}, об'єднані горизонтально, еквівалентної матриці W, що була раніше зазначена.

Приведені рівняння опускають додатковий вектор зміщення для стислості. Також, варто зауважити, що нелінійність представлену тангенсом гіперболічним, можна замінити ReLU. Якщо вхідні вектори x_t мають розмірність *D*, а розмір прихованих станів *H*, то *W* є матрицею розміру (H × (D + H)). Інтерпретуючи рівняння, нові приховані стани на кожному кроці в момент часу є лінійною функцією елементів x_t, h_{t-1}.

Vanilla RNN має просту форму, але, адитивні взаємодії є слабкою формою зв’язку між входами та прихованими шарами. Функціональна форма Vanilla RNN призводить до небажаної динаміки під час зворотного поширення, зокрема, градієнти, як правило, або зникають, або вибухають протягом тривалих періодів часу. Проблему вибухаючого градієнта можна полегшити за допомогою евристики – відсікання градієнтів при якомусь максимальному значенні, але RNN більш вразлива до зникаючого градієнта.

Рекурентна довга-короткочасна пам’ять (long-short term memory, LSTM) призначена для усунення обмежень Vanilla RNN. В даному випадку

формула рекуррентності має форму, яка дозволяє вхідним даних x_t та h_{t-1} взаємодіяти шляхом мультиплікативної взаємодії, а рекурентна LSTM використовує адитивні взаємодії в часі, які ефективніше поширюють градієнти назад у часі. На додаток до вектора прихованого стану h_t, LSTM також підтримують вектор c_t пам'яті.

$$(i, f, o, g)^T = (sigm, sigm, sigm, \tanh)^T W(x_t, h_{t-1}) \tag{1.6}$$

$$c_t = f \odot c_{t-1} + i \odot g \tag{1.7}$$

На кожному часовому кроці LSTM може вибрати для читання, запису або регулювання елемент за допомогою механізмів, які називаються gating mechanisms.

$$h_t = o \odot \tanh(c_t) \tag{1.8}$$

Тобто тут сигмоїдальна функція sigm і tanh застосовується поелементно, і якщо вхідна розмірність *D*, а прихований стан має *H* одиниць, то матриця *W* має розміри (4H × (D + 3)). Три вектори $i, f, o \in R^H$ контролюють, чи оновлюється кожна клітинка пам'яті, чи скидається вона до нуля і чи виявляється її локальний стан у прихованому векторі відповідно. Активація цих векторів базується на сигмоїдальній функції і, отже, дозволяє плавно коливатися від нуля до одиниці, щоб модель не була диференційованою.

Вектор $g \in R^H$ знаходиться в діапазоні від -1 до 1 і використовується для додаткової модифікації вмісту пам'яті. Так адитивна взаємодія є важливою особливістю конструкції LSTM, оскільки під час зворотного розповсюдження операція суми просто порівну розподіляє градієнти, що

дозволяє градієнтам на клітинках пам'яті рухатись назад через час безперервно протягом тривалих періодів часу, або принаймні до тих пір, поки потік не порушиться при мультиплікативній взаємодії активного забуття.

Таким чином, приведені системи, а саме Vanilla RNN та LSTM досить активно використовуються для обробки природних мов. Рекурентна нейрона мережа допомагає більш ефективно обробляти дані, що поступають на вхід системи, а LSTM нівелює певні обмеження архітектури RNN. Така комбінація архітектур задовольняє обмеження, коли на вхід може подаватися не тільки один, а декілька векторів даних. Вирішити цю проблему може саме рекурентна нейронна мережа.

1.5 Постановка задачі дослідження

Задачі розуміння, пізнання вирішуються з використанням в першу чергу зорових та звукових сигналів. Зір та мова є основними каналами сприйняття світу і тому при побудові когнітивних сервісів дуже важливо, використовувати методи, які дозволяли б зв'язувати інформацію між двома каналами сприйняття, а не обробляти кожен потік інформації самостійно.

Такий підхід дозволить комплексно та більш ефективніше вирішувати цілий ряд актуальних задач. По-перше, використання природної мови як простору міток для візуального розпізнавання має багато переваг. Мова – це кодування, яке було сформовано природнім чином і може представляти іменники (предмети, людей, сцени), прикметники (атрибути), дієслова (дії), що формують складні конструкції, які позначають відносини. Тому прогнозування висловлювань на природній мові узагальнює та успадковує завдання розпізнавання зображень, які в даний час розглядаються у окремій сфері комп'ютерного зору.

По-друге, кінцевими користувачами систем комп’ютерного зору є люди, які вже вільно володіють природною мовою. Таким чином, необхідно щоб системи штучного інтелекту використовували мову як простір міток, які могли б надати значно простіші та природніші взаємодії між комп’ютерами та користувачами, не вдаючись до перекладу між природною мовою та фіксованими категоріями на вході чи виході систем. Наприклад, пошук в особистій колекції фотографій із довільними текстовими запитами, такими як «фотографії, на яких я плаваю з друзями біля катеру», не повинен проходити проміжні етапи виклику класифікаторів дій для «плавання» або класифікаторів об’єктів для «особи» або «катер». Також, навпаки, комп’ютер міг безпосередньо описати оточення людині з вадами зору або відповісти на запитання про оточуюче середовище. Проте цей підхід містить певні проблеми, однією з них є те, що оцінювання стає складнішим. Наприклад, при вирішенні задачі класифікації зображень для кожного зображенню встановлюється відповідність із певною категорією. При проведенні класифікації з використанням обраної архітектури обчислюється точність розпізнавання, тобто лише частина результатів буде правильною. Однак, якщо модель маркує зображення текстом, то може бути складніше оцінити, наскільки правильним є цей текст. Потрібен набір пар позначок-речень, написаних людьми, з якими можуть бути порівняні результати класифікації.

Завдання даної роботи полягає у поєднанні візуального розпізнавання з елементами розпізнавання мови. Спільне вирішення цих завдань є актуальними, оскільки дозволить сформулювати єдину модель, яка автоматично виявляє всі свої проміжні уявлення під час навчання, без необхідності чітко вказувати, які конкретні особливості слід виділити під час обробки зображення для підтримки завдання опису.

Мета кваліфікаційної роботи полягає у поєднанні методів розпізнавання зображень та тексту для того, щоб промаркувати

зображення реченнями, що відображають сенс цих зображень. Таке маркування дає можливість побудувати когнітивний сервіс, що одночасно працює і з зображеннями і з текстом.

В магістерській роботі вирішуються такі задач:

– аналіз характеристик когнітивних систем та сервісів;

– дослідження задач комп’ютерного зору та аналіз підходів до розпізнавання образів;

– дослідження методів обробки природних мов;

– розробка методу маркування зображень текстом;

– експериментальна перевірка методу маркування зображень текстом.

2 МАРКУВАННЯ ВІЗУАЛЬНИХ ОБ'ЄКТІВ ТЕКСТОМ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

2.1 Метод маркування зображень текстом

У даній роботі розглядається задача маркування набору зображень, що надходять у необробленому вигляді у інформаційну систему. В якості вхідних даних надається кінцева множина зображень, кожному з яких додається текстовий опис. Тобто дані містять пару зображення – текст (речення). В результаті виконання маркування необхідно повернути оцінку їх сумісності. Іншими словами, для кожного із речень, необхідно ранжувати зображення на основі того, наскільки добре вони зображують це речення. Альтернативна задача полягає в тому, щоб для любого зображення із вхідної виборки ранжувати речення у відповідності до того, наскільки добре вони описують це зображення.

При вирішенні цієї задачі виникає певна проблема, яка полягає в тому, що зображення і речення є складними об'єктами великого розміру. Тому взаємозв'язок між цими об’єктами потребує одночасної обробки зображень та тексту, а також встановлення формального зв'язку між ними. Також треба враховувати обмеження налаштування для речень-маркерів. На практиці при встановленні такого зв'язку для кожного зображення можна отримати множину схожих речень.

Метод маркування візуальних об'єктів текстом складається з трьох ключових етапів:

– кодування зображення у вектор;

– кодування тексту у вектор;

– порівняння векторів.

При кодуванні зображення у вектор використовується згорткова нейронна мережа і зображення розбивається на фрагменти, які й кодуються. Тому розмір вектору для зображення можна зафіксувати.

При кодуванні тексту у вектор визначається вага слів у тексті реченні в залежності від частоти їх появи у словнику, який використовується для кодування. Далі речення кодується, наприклад, як сума кодів слів у його складі.

При порівнянні оцінюється схожість між вектором зображення та вектором тексту (речення).

Порівняння виконується таким чином: речення у парі із «справжніми» зображеннями повинні мати високий бал, а речення у парі із «хибними» зображеннями мають низький бал.

Для маркування візуальних об'єктів необхідно сформувати єдину нейронну мережу, яка виконує приведені завдання в рамках наскрізної (end-to-end) парадигми навчання і без використання явних уявлень про ознаки. Такий підхід покращує ефективність, оскільки всі компоненти моделі мають одну і ту ж мету. Також модель поповнюється відповідностями саме між окремим фрагментами зображень та речень, а не глобальними об’єктами. Це підвищує продуктивність маркування, а також інтерпретацію результатів.

Підхід до вирішення цієї здачі має орієнтованість на дані, зокрема модель складається з нейронної мережі, що бере одне зображення та одне речення та обчислює скалярну оцінку, вказуючи, наскільки зображення та речення співпадають.

Модель навчається на наборі зображень із реченнями і призначає високі оцінки відповідним парам зображення-речення у вхідній виборці, а низькі – при невідповідності речення та зображення.

Наприклад, на рисунку 2.1 зображена нейронна мережа, яка бере зображення та речення та обчислює скалярну оцінку відповідності.

Рисунок 2.1 – Принцип роботи системи, що розглядається

Архітектура мережі для маркування зображення текстом складається з трьох компонентів:

– кодер зображення, який кодує кожне зображення в єдиний вектор i або набор векторів фрагментів $\{i\}$ цього зображення;

– кодер тексту (наприклад, речень, що описують зображення), який обробляє речення в єдиний вектор d, або набор векторів фрагментів $\{d_m\}$;

– модуль, обчислення результуючої оцінки, що визначає ступінь схожості між зображення та текстом.

Під час навчання виконується обчислення пар балів для відповідних пар зображення – речення для кожного елементу в наборі навчальних даних. Функція втрат задається так, щоб оцінки пари справжнє зображення – речення, були вищими, ніж оцінки пари неправдиве зображення-речення.

На рисунку 2.2 наведено приклад оцінок для трьох зображень. Для пар справжнє зображення – речення оцінки схожості зображено зеленим кольором, а для пар неправдиве зображення – речення ці оцінки зображені червоним кольором.

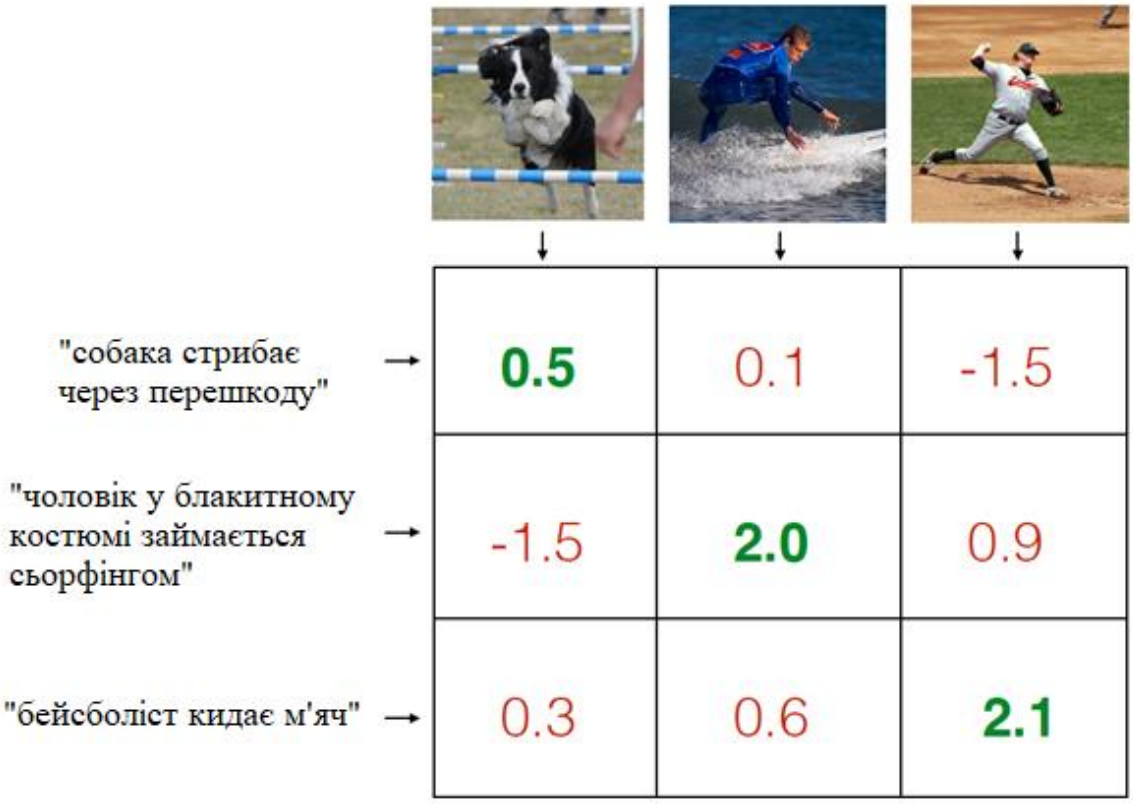

Рисунок 2.2 – Матриця оцінок відповідностей між реченням та зображенням

З рисунку 2.2, наприклад, видно, що оцінка схожості для першого речення «собака стрибає через перешкоду» та зображення собаки значно вища, ніж оцінки для альтернативних зображень із людьми.

Таким чином, маркування візуальних об'єктів текстом об'єднує у собі паралельну обробку двох типів даних – текстових та візуальних, з використанням таких підходів:

– використання згорткових (конволюційних) нейронних мереж (CNN) [18], [20] як потужного класу моделей для вирішення задачі класифікації зображень та виявлення об'єктів на цих зображеннях;

– використання рекурентних нейронних мереж для моделювання мовного контексту [15], [16], [17];

– попереднє оброблення набору речень для пониження розмірності цих даних.

Згорткові нейронні мережі містять згорткові та агрегувальні шари. Перші згортають зображення із декількох пікселів, тим самим імітуючи реакцію нейрону людина на зоровий сигнал. Другі об’єднують виходи із згорткового шару. Згорткова нейронна мережа розбиває зображення на окремі фрагменти однакового розміру і потім обробляє інтегровану інформацію з цих фрагментів. Така розбивка дає можливість знайти на зображенні характерні елементи.

Рекурентні нейронні мережі формують граф, який має орієнтацію у часі. Така мережа має внутрішню пам'ять. Ця пам'ять використовується для обробки входів.

2.2 Підготовка зображення до маркування

Як було зазначено у підрозділі 2.1, маркування зображень містить у собі обробку як зображень, так і речень на рівні фрагментів. Перед маркуванням виконується попередня обробка даних. У даному підрозділі розглядається передобробка зображень.

Оскільки при порівнянні речень та зображень необхідно враховувати як ці дані в цілому, так і їх фрагменти, то використовують дві стратегії кодування зображень:

– глобальне кодування зображень – це кодування у набір векторів $\{i_r\}$;

– кодування зображень на рівні фрагментів – кодування зображення P у вектор i.

Варто зазначити, що для підтримки популярного методу оптимізації – зворотного поширення, функція кодування містить диференційовані компоненти.

Перший підхід, глобальне кодування зображень, перетворює початкове зображення у вектори. Вони підтверджують можливість згорткових нейронних мереж [18] перетворювати «сирі» (необроблені) дані, що подаються на вхід системи у представлення [19], [20], [21], які підтримують (у середньому) ефективність людського рівня розпізнавання зображень[19]. Саме тому використовуються згорткові нейронні мережі, які спочатку попередньо навчається за допомогою системи ImageNe, де ImageNet – це набір зображень, організований відповідно до ієрархії WordNet (база даних семантичних відношень між словами більше ніж на 200 мовах). Кожне поняття в WordNet, може бути описане кількома словами або словосполученнями, називається «набором синонімів». У WordNet понад 100 000 таких наборів; більшість із них – іменники (80 000). Зображення кожної концепції контролюються якістю та коментуються людиною.

Потім видаляється останній шар згорткової нейронної мережі, який обчислює 1000 ймовірностей різних класів системи ImageNet, проте зберігає всі інші шари та параметри в цілості. Приведена процедура розглядається в якості вилучення ознак з даних, а саме використовується функція-екстрактор $E_\beta(P)$ яка приймає на вхід піксельне зображення *P* та має певний параметр β. Наприклад, AlexNet [20] має приблизно 60 мільйонів параметрів β, а $E_\beta(P)$ – 4096-мірний вектор, який представляє собою поєднання подальшої нелінійності (наприклад, ReLU в AlexNet) і безпосередньо перед класифікатором ImageNet, який відкидається.

На практиці досить часто використовується попередньо навчена CNN як екстрактор фіксованих ознак та їх обчислення для всіх зображень у наборі даних.

Тоді кодування зображення *i* набуває вигляду:

$$i = W\left(E_\beta(P)\right) + b. \tag{2.1}$$

де матриця W – матриця коефіцієнтів зображення.

Тобто зображення кодується, беручи вектор ознак зображення і передаючи його через лінійне перетворення. Таким чином навчаються параметри W та b і вектор i продовжить подальшу обробку в мережі.

Також варто зазначити, що замість цього можна легко також здійснити зворотне розповсюдження через CNN і відрегулювати параметри β. Такий процес називається доопрацювання (finetuning), незважаючи на те, що він концептуально простий, у багатьох випадках такий підхід може бути недостатньо важким в обчисленнях (CNN часто використовує більшу частину обчислень) і практично складним порівняно з простим посиланням на попередньо обчислений 4096-мірний вектор для будь-якого зображення.

Наступним використовується підхід «Кодування зображень на рівні фрагментів». Об’єкти виявляються на кожному зображенні за допомогою згорткової нейронної мережі, що орієнтується на регіони зображення – Region Based Convolutional Neural Networks (R-CNN).

Обрана згорткова нейронна мережа проходить попередньо тренується на ImageNet [21], а далі доопрацьовується на 200 класах програми ImageNet Detection Challenge [19]. Далі використовується 19 найкращих виявлених місць на додаток до всього зображення, тобто 20 областей загалом, і обчислюється представлення на основі пікселів P_r всередині кожної області обмежуючого поля з:

$$i_r = W\left(E_\beta\left(P_r\right)\right) + b, \tag{2.2}$$

де, $E\ (I_r)$ перетворює пікселі всередині обмеженої області I_r, наприклад в 4096-вимірні активації повністю пов'язаного шару безпосередньо перед класифікатором. Матриця W має розміри $h \times 4096$, де h – розмір мультимодального вкладеного простору. Таким чином, кожне зображення представляється як набір h-мірних векторів i_r, де r = 1.. . 20.

Треба зауважити, що у цьому кодуванні використовується зовнішній, попередньо навчений та фіксований детектор об’єктів для ідентифікації 19 областей, які слід кодувати. Проте це не бажано, оскільки отриманий підхід не є повністю end-to-end підходом, але його використання спрощує процесс обробки та понизити складність обчислень.

Тобто, нейронні мережі, які відносяться до класу згорткових нейронних мереж мають змогу кодувати зображення у вектор i або набір векторів фрагментів $\{ i \}$ з функціями, що складаються з диференційованих компонентів та параметрів.

2.3 Підготовка текстових даних до маркування зображень

Наступним кроком є передобробка даних, що надходять у систему у вигляді речень. Вони мають вигляд послідовності слів, з яких складається речення, і необхідно представити її як один вектор d або набір векторів фрагментів d_m:

$$d = \{d_m\}. \tag{2.3}$$

Також припускається, що кожне речення представлене у вигляді послідовності слів, взятих із фіксованого словника D усіх можливих слів. Кожне слово кодується з використанням так званого гарячого кодування, або one-hot encoding. За цим підходом усі значення елементів вектору d_m

дорівнюють нулю, за винятком одиниці в рядку з індексом слова, що присутній у словниковому запасі. Гаряче кодування *m*-го слова у реченні із послідовності закодованих векторів $(C_1,...,C_M)$, де *M* це номер слів має вигляд:

$$C_m \in R^{|D|}. \tag{2.4}$$

Також, варто зазначити, що на практиці деякі слова можуть бути дуже рідкісними, або взагалі не зустрічатися у навчальних даних; у цьому випадку загальноприйнятим є додавання спеціального слова unk (скорочення від «невідомо») до словникового запасу та переформатування всіх рідкісних слів, наприклад, що трапляються лише 5 разів у навчальних даних, до слова unk під час попередньої обробки.

Одним з найпростіших способів кодування речення є представлення кожного слова як окремого фрагменту речення та обчислення його представлення за допомогою one-hot кодування, вектор C_m з лінійним перетворенням:

$$d_m = W^* C_m \tag{2.5}$$

де W^* – матриця параметрів, яка буде навчатися під час зворотного поширення.

Оскільки C_m – це кодований one-hot encoding вектор, ця операція ефективно вибирає один рядок матриці W^* – матриці вбудовування слів (the word embedding matrix).

У випадку, якщо виникає перенавчання через відсутність достатньої кількості даних, матриця W^* також може бути реалізована для векторів

слів, отриманих з інших завдань з навчанням без учителя, таких як word2vec [22].

Окремі репрезентації слів можуть бути фрагментами речення, або вони можуть бути об`єднаними в одне речення або через середнє значення, або через суму, тобто:

$$d = \sum_m d_m. \tag{2.6}$$

Таке кодування може представляти такі слова, які трапляються десь у реченні, але їх просторові зв'язки всередині речення втрачаються під час кодування. Наприклад, кодуючи речення «червона чашка на дерев'яному столі», в такому випадку зберігається основна інформація, проте втрачається інформацію про те, які слова описують предмети «червоний» чи «дерев'яний».

Приведений підхід можна розширити, кодуючи не окремі слова, а біграми (пари) або триграми (трійки) суміжних слів, а більш загальний термін – *n*-грамовими представленнями. Найпростішим підходом може бути об'єднання окремих слів в один вектор d_l:

$$d_l = d_m + d_{m-1}. \tag{2.7}$$

Також можна обробити цей об'єднаній вектор через один або два шари повністю пов'язаної нейронної мережі, що дозволить мережі розрізняти та представляти наявність «червоної чашки» замість «червоної» та «чашки» окремо.

Результатом (2.6) є представлення речення із декількох слів. Однак дане представлення розглядає речення як множину слів і не враховує зв'язок між словами.

Рекурентні нейронні мережі кодують речення, створюючи представлення, яке є функцією всіх слів у реченні та всіх їх взаємозв’язків. Загальноприйнятим підходом є кодування кожного слова спочатку за допомогою матриці вкладання слів, а потім подання по одному кожне закодоване слово в RNN.

Конкретно, ми обчислимо для m = 1, ..., T:

– ініціалізація прихованих станів RNN при векторі нулів представлена формулою:

$$g_0 = \vec{0}, \tag{2.8}$$

– обчислення вкладених слів:

$$e_m = W^* C_m, \tag{2.9}$$

– об’єднання всіх представлень слів за допомогою Vanilla RNN.

Нарешті, представлення d для всього речення може бути або останнім прихованим станом RNN

$$d = g, \tag{2.10}$$

або сумою або середнім значенням усіх прихованих станів:

$$d = \sum_{m=1}^{M} g_m. \tag{2.11}$$

Рекурсивні нейронні мережі – це простий кодер речень, який має наступні переваги в рамках поставленої задачі:

– можливість кодування всіх слів та їх взаємозв'язків;

– дуже ефективні у реальних задачах, оскільки система сканує лінійно по реченню.

Однак можна стверджувати, що речення за побудовою мають композиційну структуру і що краще підходить кодер буде додержуватися цієї ієрархічної структури під час своєї роботи.

Рекурсивні нейронні мережі є узагальненням рекурентних нейронних мереж, оскільки вони дозволяють більш складне кодування над довільною структурою дерева. У свою чергу, Vanilla RNN – це особливий випадок, коли дерево є ланцюжком.

Через те, що задача визначення структури дерева для кожного речення є нетривіальною, тому досить часто використовується алгоритм синтаксичного розбору в реченні для визначення порядку злиття. Також іншим недоліком рекурсивних нейронних мереж є той факт, що такі мережі важче розпаралелювати, оскільки кожне речення має різний спосіб злиття.

Основна ідея полягає в тому, щоб розпочати з векторів слів в якості листків дерева, а потім рекурсивно об’єднати їх представлення аж до кореневого вузла дерева, який стає представленням для всього речення. Тобто, обчислюється представлення g_n для деякого вузла у дереві n як:

$$g_n = u\left(g_{t_n^1}, ..., g_{t_n^q}\right). \tag{2.12}$$

де t_n^1 та t_n^q є дочірніми вузлами вузла n,

u є функцією злиття.

Базовим випадком є окремі слова, для яких приховане представлення обчислюється за допомогою матриці вкладання слів, як і раніше. Для дерева синтаксичного аналізу, яке організовує ієрархію речення у бінарне дерево, функція злиття Continious-Time рекурентна нейронна мережа (CTRNN) може мати вигляд:

$$g_n = f\left(W_l^* g_{t_n^1} + W_r^* g_{t_n^2}\right). \quad (2.13)$$

де f – нелінійність; типовими налаштуваннями функції активації f є або ReLU, або tanh. (наприклад, tanh);

W_l^*, W_r^* - матриці параметрів,

t_n^1, t_n^2 - ліве та праве дочірні листки відповідно.

Deep-Tree Recursive Neural Network (DTRNN) [23] натомість об’єднує представлення над деревом залежностей, яке упорядковує слова в структурі дерева, пов’язані між собою позначеними ребрами, що описують синтаксичні зв’язки. Дерево залежностей має ту перевагу, що словам, які мають головне значення, такі як основна дія або дієслово, система надає більший вплив на представлення остаточного речення.

Формула злиття має дуже подібну форму до попередньої формули, за винятком того, що вміщує кілька можливих дочірніх елементів на вузол.

2.4 Побудова функції втрат для порівняння зображень та тексту

Використовуючи розглянуті декілька підходів для обчислення або глобального представлення, де кожне зображення кодується у вектор *i*, і кожне речення у вектор *d*, або уявлення фрагментів, де кожне зображення кодується у набір векторів *{ i }*, а кожне речення у набір векторів *{ d }*. Наступним кроком є дослідження функцій втрат, які можуть вирівняти вектори двох модальностей, припускаючи, що ці вектори займають загальний мультимодальний простір представлення.

Припустимо, що ми маємо навчальний набір із *N* образів та речень. Використовуючи глобальні кодери зображень і речень, ми можемо

обчислити різницю між векторами i_n для зображень та d_n для тексту. Кожен елемент першого з цих векторів містить образ зображенння, а другого – образа тексту.

Для оцінки схожості цих векторів можна використати суму квадратів відхилень:

$$L = \sum_{n=1}^{N} \left\| i_n - d_n \right\|^2 . \tag{2.14}$$

Представлена функція витрат має мінімальне значення у випадку близькості зображень та речень. Однак ця функція не дасть коректну оцінку у випадку нульових векторів. В даній ситуації перед оцінкою потрібно використати попередню обробку та вилучити такі вектори із множини даних, що оцінюються.

Більш підходяща альтернатива, яка також працює набагато краще на практиці [23], [24], може бути сформульована в рамках максимального відступу (max-margin).

Зокрема, доцільним є використання внутрішнього добуток $S = i^T d$ між зображенням та вектором речення як оцінку, що вказує на ступінь сумісності. Оскільки:

$$i^T d = \|i\| \|d\| \cos(\theta) \tag{2.15}$$

точковий добуток пов'язаний з кутом між двома векторами, припускаючи, що вони обидва мають однакову довжину. Отже, цей вибір моделювання має інтерпретацію вбудовування як зображень, так і речень у загальний мультимодальний простір та сприйняття двох об’єктів як сумісних, якщо вони спрямовані в подібному напрямку.

Альтернативною, більш конкретною, інтерпретацією розмірів *i, d* полягає в тому, що вони можуть навчитися виявляти пов'язані ознаки за двома способами. Наприклад, перший вимір *i* може бути позитивним, якщо на зображенні є фрагменти, схожі на хутро кішки, а перший вимір *d* може бути позитивним, якщо десь у реченні було згадано «кішку» або якусь іншу пухнасту тварину. Тоді, якщо обидва ці виміри є або позитивними, тобто обидва присутні, або обома негативними – обидва відсутні, вони будуть позитивно взаємодіяти в точковому добутку та збільшуватимуть бали відповідності. І навпаки, якщо на зображенні виявлені текстури хутра, але у реченні не згадуються тварини, або слова схожі на хутро, показник відповідності зменшиться.

У цілому, якщо значення оцінки співпадіння $D = i^T d$ позитивна, то такі пари речень-зображень правильні, і навпаки, а неправильні пари повинні мати негативні оцінки. У випадку, якщо є один вектор зображення *i* і вектор речення *d*, можна обчислити оцінку відповідності як внутрішній добуток $D = i^T d$.

Також, існує певна асиметрія:

– все, що згадується у реченні, повинно знаходитись на зображенні, якщо оцінка має бути високою;

– зображення можуть містити багато інших об'єктів, які можуть не згадуватися в підписах, оскільки вони мають низьку оцінку.

Проте ця особливість не повинна заважати системі призначати високий бал відповідності, навпаки, ця асиметрія спонукає обчислювати оцінку відповідності пари зображення-речення як функцію оцінки фрагмента зображення-речення наступним чином:

$$D = \sum_m \max_j i_j^T d_m. \qquad (2.16)$$

Тобто, для кожного фрагмента речення d_m і $\max i^T$ оцінює найкращу відповідність цього фрагмента будь-якому із фрагментів зображення. Потім P t додає, наскільки кожен фрагмент речення вирівнюється з чимось на зображенні, щоб обчислити остаточну оцінку відповідності.

2.5 Процес навчання системи маркування зображень

Пошук екстеремуму, що відповідає мінімальній розбіжності між реченнями та зображеннями, виконується на двох рівнях:

– градієнтий спуск як внутрішня оптимізація;

– оптимізація гіперпараметрів.

Використовуються такі методи:

– градієнтний спуск;

– крос-валідація;

– регуляризація.

Для оптимізації моделі використовується стохастичний градієнтний спуск (Stochastic Gradient Descent, SGD). Градієнтний спуск полягає у знаходженні локального екстремума функції при переміщенні по градієнту, тобто у напрямку максимального зменшення функції втрат. При стохастичному градієнтому спуску параметри моделі обчислюються заново для кожного елемента вхідної виборки.

Додатково виконується крос-валідація (перехресна перевірка). Процедура такої перевірки полягає в тому, що вхідна вибірка розбивається на декілька підмножин. Однак із підможин використовується для тестування, всі інші – для навчання. Цикл навчання повторюється згідно кількості підмножин, тобто кожна підмножина використовується для перевірки. Оцінка, отримана в результаті крос-валідації, рівномірно використовує вхідні дані. Дана операція є важливою, оскільки вона дозволяє перевірити рівень навчання та ступінь регуляризації.

Регуляризація призначена для боротьби з перенавчанням. Перенавчання виникає у випадку, коли модель дуже точно відповідає вхідній виборці та не є узагальненою. Регуляризація полягає у призначенні штрафу за великі значення коефіцієнтів в моделі. В результаті «згладжується», узагальнюється модель, та вирішується проблема перенавчання.

Оптимізація зі стохастичним градієнтом може розглядатися як внутрішній цикл оптимізації, тоді як оптимізація гіперпараметрів – це зовнішній цикл, який визначає хороші значення гіперпараметрів, які важко або неможливо повернути назад.

Значення гіперпараметрів використовуються для змін у процесі навчання. Їх підбирають для кожної моделі так, щоб найбільш ефективно навчити модель. Приклад гіперпараметру: швидкість навчання. При збільшенні кроку навчання можна «проскочити» екстремум функції втрат. Малий крок приводить до занадто довгого навчання.

Приведений процес складається з вибірки гіперпараметрів з певного діапазону пошуку, оптимізації моделі та оцінки моделі у складі перевірки. Остаточна найкраща модель – це та модель, яка досягає найкращих показників на тестовій виборці.

Після того, як визначається найкраще навчена модель, що має найменші втрати при валідації, модель оцінюється один раз на наборі тестів та робиться висновок про її результативність.

Послідовні вдосконалення завжди можна отримати, використовуючи ансамблі моделей, які усереднюють результати оцінки декількох моделей, навчених з різних ініціалізацій або з різними гіперпараметрами.

3 ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА МЕТОДУ МАРКУВАННЯ ЗОБРАЖЕНЬ ТЕКСТОМ

У цьому розділі розглядаються результати експериментальної перевірки методу маркування. Метод реалізовано у вигляді гібридної нейронної системи. Оцінюється система, яка використовує кодер фрагментів зображення та двонаправлений кодер фрагментів BRNN. Для проведення експериментальної перевірки було використане наступне програмне забезпечення: мова програмування python, а також такі бібліотеки та фреймворки, як TensorFlow 2, pandas, numpy та Keras.

3.1 Робочий процес нейронної мережі

Перший етап перевірки – це підготовка даних. В рамках цієї роботи припускається, що на вхід системи надходить на набір даних, що складається з набору пар (x, y), де *x* є деяким прикладом з набору даних, а *y* - міткою.

Потім обраний набір даних розділяється на три частини, як правило, тренувальну, перевірочну та тестову, у свою чергу загальні пропорції можуть становити відповідно 80%, 10%, 10%.

В даній роботі навчальна вибірка використовується для оптимізації параметрів з допомогою зворотного поширення помилки, перевірочна вибірка – для оптимізації гіперпараметрів, та тестовий набір даних для оцінки.

Попередня обробка даних може допомогти поліпшити збіжність нейронних мереж [18]. Для зображень загальні методи попередньої обробки передбачають стандартизацію даних, наприклад, віднімання середнього та ділення на стандартне відхилення індивідуально для кожного вхідного виміру *x*, або принаймні віднімання середнього.

Наступним кроком визначається сімейство архітектур, тобто проектування внутрішніх частин обчислювального графа, що формує функцію *f*. На цьому етапі обговорюються кілька типових евристик, що використовуються на практиці. Є загальноприйнятим обробляти дані, що переставлені пікселями за допомогою конволюційних нейронних мереж та послідовності даних – рекурентними нейронними мережами.

Стосовно масштабу архітектури, то застосовується загальне правило: повна модель повинна мати приблизно таку ж кількість параметрів, скільки є прикладів в навчальному наборі даних. Наприклад, набір даних ImageNet [19] містить 1 мільйон прикладів або невеликий коефіцієнт більше, якщо врахувати збільшення даних. Згорткові нейронні мережі, які тренуються на ImageNet, зазвичай мають порядку 10M параметрів, і методи регуляризації, такі як регуляризація L2, випадання та збільшення даних, використовуються для подальшого обмеження моделі для запобігання перенавчанню.

Оптимізація гіперпараметрів, як і цикл мінімізації функції втрат, може бути реалізована за допомогою стохастичного градієнтного спуску. Дана оптимізація розглядається як зовнішній цикл, який визначає найкращі значення гіперпараметрів.

Процес складається з пошуку гіперпараметрів з певного, попередньо визначеного, діапазону пошуку, для подальшої оптимізації та оцінки моделі. У результаті визначається найкраща модель, яка досягає найкращих показників валідації.

Після визначення найкращої моделі, вона оцінюється на тестовому наборі даних й виводить кінцевий результат. Також, використовуючи ансамблі моделей, можна пошарово вдосконалювати модель, усереднюючи результати оцінки декількох моделей, навчених з різних початкових даних або з різними гіперпараметрами.

3.2 Опис вхідних наборів даних

Проблема роботи полягає у маркуванні зображення із відповідним реченням, яке містить точний опис вмісту рисунку. В рамках даної задачі була розглянута нейронна мережа, яка може паралельно обробляти, як набір даних, який містить текстову інформацію, так і дані, що представлені у вигляді зображень. Саме тому були обрані набори даних із реченнями і зображеннями Flickr8K [25] та Flickr30K [26].

Flickr8K – це набір даних для опису та пошуку зображень на основі речень, що складається з 8000 зображень, кожна з яких поєднана з п’ятьма різними заголовками, що забезпечують чіткий опис основних сутностей та подій. Зображення були обрані з шести різних груп Flickr і, як правило, не містять жодних відомих людей чи місць, але були обрані вручну, щоб зобразити різноманітні сцени та ситуації. Представлений набір даних містить два стовпці, де перший – це назва файлу зображення, а другий – його опис. Приклад цих даних зображений на рисунку 3.1.

	image	caption
0	1000268201_693b08cb0e.jpg	A child in a pink dress is climbing up a set o...
1	1000268201_693b08cb0e.jpg	A girl going into a wooden building .
2	1000268201_693b08cb0e.jpg	A little girl climbing into a wooden playhouse .
3	1000268201_693b08cb0e.jpg	A little girl climbing the stairs to her playh...
4	1000268201_693b08cb0e.jpg	A little girl in a pink dress going into a woo...

Рисунок 3.1 – Приклад даних з набору даних Flickr8K

У свою чергу, опис зображений на рисунку 3.2, а саме зображення на рисунку 3.3.

```
A child in a pink dress is climbing up a set of stairs in an entry way .
A girl going into a wooden building .
A little girl climbing into a wooden playhouse .
A little girl climbing the stairs to her playhouse .
A little girl in a pink dress going into a wooden cabin .
```

Рисунок 3.2 – Приклад опису одного зображення з набору даних Flickr8K

Рисунок 3.3 – Приклад зображення з набору даних Flickr8K

Набір даних Flickr30K, так як Flickr8K призначений для опису зображень на основі речень. У цьому наборі даних представлені 31000 зображень і відповідні підписи, які пов'язують згадування одних і тих самих об'єктів(наприклад, дівчина) у різних підписах для одного зображення та асоціюючи їх іншими. Приклади даних зображені на рисунках 3.4 та 3.5.

Варто зазначити, що кожне зображення в обох наборах даних анотоване 5 реченнями, які були написані працівниками Amazon Mechanical Turk (AMT).

```
Two young guys with shaggy hair look at their hands while hanging out in the yard .
Two young , White males are outside near many bushes .
Two men in green shirts are standing in a yard .
A man in a blue shirt standing in a garden .
Two friends enjoy time spent together .
```

Рисунок 3.4 – Приклад опису одного зображення з набору даних Flickr30K

Пропорції, в яких були розділені обидва набори даних наступні: використовується 1000 зображень для перевірки, 1000 для тестування та решту для навчання.

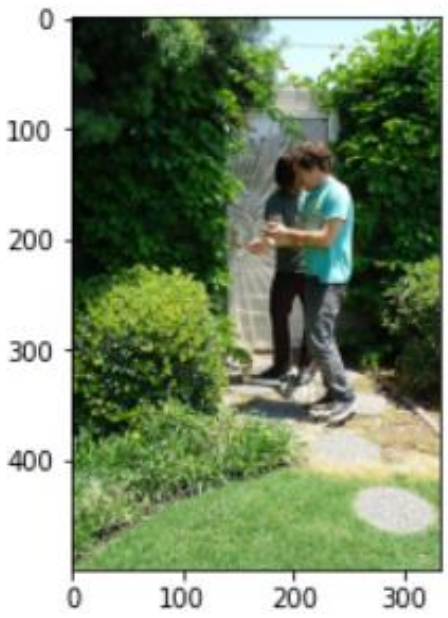

Рисунок 3.5 – Приклад зображення з набору даних Flickr30K

Для попередньої обробки даних усі речення переводяться у строчний регістр та відкидаються нелітерально-цифрові символи за допомогою відповідних функцій у мові python. Наступним кроком є фільтрування слів. Ті, слова, що трапляються менше 5 разів у навчальному наборі

відкидаються для полегшення процесу навчання, що дає словники із 2538 та 7414 слів для наборів даних Flickr8K та Flickr30K відповідно.

3.3 Оцінка метрики

Для оцінки моделі був взятий тестовий набір зображень і речень і у результаті отримані елементи одного типу в результаті запиту для елементу другого типу. Зокрема, враховуючи зображення, необхідно, щоб його «рідний» підпис займав високе місце у списку всіх тестових речень, відсортованих за оцінкою S – це задавання анотації зображень, і навпаки, отримуючи відповідне речення, необхідно, щоб його зображення отримало високе місце у списку всіх тестових зображень, відсортованих за оцінкою S – це завдання пошуку зображень.

Тому, для кожного елемента приписується ранг r із цілочисельним зазначенням положення його правильного асоційованого елемента в рейтинговому списку.

В абсолютно ідеальній ситуації повинна бути 1 для всіх зображень та речень, що вказує на те, що правильним результатом є верхній елемент у списку.

Потім визначається середній ранг по всіх елементах та вимірюється частота випадків, коли правильний елемент був знайдений серед найкращих результатів. Через те, що кожне зображення має 5 речень, в при маркуванні зображення фіксується найкращий (найнижчий) ранг серед 5 правильних речень, так що найкращий середній ранг залишається рівним 1. Приклад пошуку показано на рисунку 3.6.

Для кожного зображення виводиться найбільш сумісне тестове речення та візуалізується область з найбільшим балом для кожного слова та пов’язаних оцінок. Слова, які мають низькі бали не виводяться на

зображення з метою структурування та компактності результату, а кожному регіону присвоюється довільний колір.

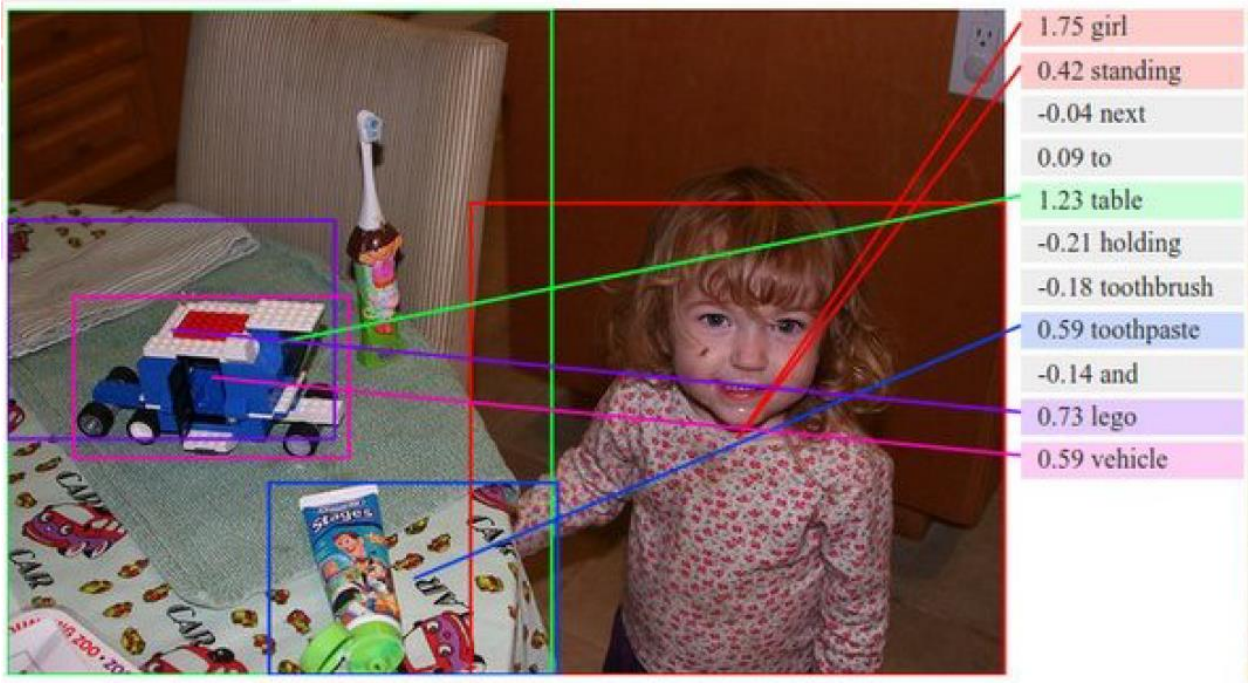

Рисунок 3.6 – Приклад маркування зображення

Результат цих експериментів показано на рисунку 3.7.

Model	Image Annotation R@1	R@5	R@10	Med r	Image Search R@1	R@5	R@10	Med r
Flickr8K								
Kiros et al. [47]	13.5	36.2	45.7	13	10.4	31.0	43.7	14
Mao et al. [65]	14.5	37.2	48.5	11	11.5	31.0	42.4	15
Our implementation of DeFrag [45]	13.8	35.8	48.2	10.4	9.5	28.2	40.3	15.6
Our model: DepTree edges	14.8	37.9	50.0	9.4	11.6	31.4	43.8	13.2
Our model: BRNN	**16.5**	**40.6**	**54.2**	**7.6**	**11.8**	**32.1**	**44.7**	**12.4**
Flickr30K								
SDT-RNN (Socher et al. [90])	9.6	29.8	41.1	16	8.9	29.8	41.1	16
Kiros et al. [47]	14.8	39.2	50.9	10	11.8	34.0	46.3	13
Mao et al. [65]	18.4	40.2	50.9	10	12.6	31.2	41.5	16
Donahue et al. [18]	17.5	40.3	50.8	9	-	-	-	-
DeFrag (Karpathy et al. [45])	14.2	37.7	51.3	10	10.2	30.8	44.2	14

Рисунок 3.7 – Результат експериментів

На рисунку 3.7 використано такі позначення. R @ K - це Recall, @ K (високий – це добре). Med r – середній ранг (низький – хороший). У результаті 5 найкращих моделей оцінюється кожна незалежно на тестовому наборі, а потім виводиться середня ефективність. Стандартні відхилення для значень відкликання коливаються приблизно від 0,5 до 1,0.

Відображення фрагментів дають вищу продуктивність. По-перше, повна модель BRNN на основі фрагментів суттєво перевершує систему запропоновану у [23] – Socher, який тренувався з однаковою втратою рейтингу, але використовував глобальний кодер зображень та глобальний кодер речень – рекурсивна нейронна мережа. Однак це порівняння є не зовсім справедливим, оскільки Socher також використали менш потужний CNN, заснований на автокодері [27], тоді як результати нашої системи використовують AlexNet [20], який відображає чудову продуктивність у завданні ImageNet [19]. Варто зазначити, що AlexNet використовується для вилучення об’єктів для цілих зображень або виявлених об’єктів та утримання фіксованих ваг під час оптимізації, тобто CNN не налаштовується, що робиться суто для практичного спрощення експериментів.

Подібну втрату в рейтингу прийняла система, яка була запропонована у [28], які використовують кодер речень LSTM. Проте їх результати перераховуються за допомогою CNN, еквівалентного за потужністю до AlexNet, але подібного до Vinyals [29] вони перевершують модель, що розглядається у цій роботі за допомогою потужнішої мережі CNN (VGGNet [30] та GoogLeNet [31] відповідно).

Однак, контролюючи ефективність CNN, можна побачити вдосконалення у використанні кодерів на рівні фрагментів та оцінок. Наприклад, 50,9→61,4 для анотації зображення R @ 10 та 46,3→50,5 для пошуку зображень R @ 10.

Система DeFrag, яка була запропонована у [24], порівняно з системою, що розглядається у цій роботі використовує різні вектори слів, дропаут для регуляризації та різні діапазони крос-валідації. Тому в рамках цієї роботи DeFrag була адаптована для того щоби мати змогу порівняти ці дві системи. DeFrag використовує більш складну функцію втрат, що складається з двох втрат (втрата фрагментів та глобальна втрата), яка була усунена на користь більш простого 2.23 та функції ранжування втрат у рівнянні 2.22.

З метою дослідження цієї змінної, була видалена BRNN і були використані дерева рішень відповідно до [24] для кодування фрагментів речень. Єдина різниця між моделлю DepTree та моделлю що була запропонована у [24] – це простіша функція втрат, і можна побачити, що розроблена тут формулювання досягає постійних удосконалень.

Підсумовуючи, порівняно з іншими роботами, які використовують AlexNets, розглянута у даній роботі модель демонструє постійне вдосконалення, яке ми можемо простежити до обробки зображень та речень на рівні фрагментів.

Оскільки представлення фрагментів витягують із зображення набагато більше необробленої інформації, можливе виникнення розриву, який може бути закритий набагато більшими розмірами вбудовування. Однак це розширення є нетривіальним, оскільки кількість параметрів у моделі також починає зростати дуже швидко, тоді як підхід до фрагментів не вводить більше параметрів, ніж метод глобального кодування.

BRNN перевершує можливості дерева залежностей, а саме найкраща серед порівняних модель, використовує BRNN для кодування фрагментів речень. Відносини дерева залежностей працюють краще, ніж використання лише окремих слів або біграм [24]. У свою чергу, можна спостерігати, що використання BRNN як кодера додатково покращує продуктивність відповідно рисунку 3.7.

Порівняно із використанням аналізатора – дерева зовнішніх залежностей, BRNN є end-to-end підходом кодування, який не покладається на обчислення, через які ми не можемо здійснити зворотне розповсюдження. Більше того, порівняно з використанням дерев рішень, BRNN має здатність кодування кожного фрагмента більш ніж на 2 словах.

3.4 Якісне оцінювання

Відповідно до рисунку 3.6, метод виявляє візуально-семантичні відповідності, що легко інтерпретуються, навіть для невеликих або відносно рідкісних об'єктів. Одним з обмежень методу є те, що він не підтримує підрахунок.

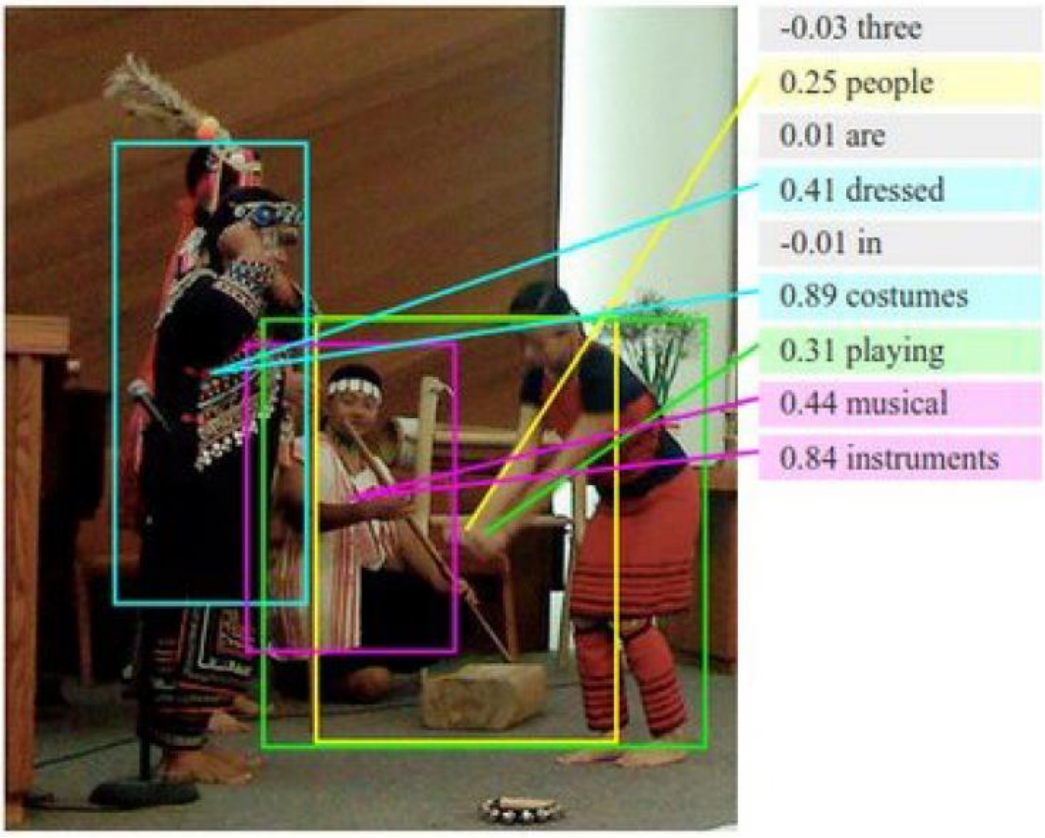

Рисунок 3.8 – Приклад маркування зображення

Наприклад, спостереження, що зображено на рисунку 3.8, містить фразу «три людини». Тобто, представлені слова повинні відповідати трьом людям на зображенні, але метод обмежується двома людьми.

Приведені результати мають обмеження, оскільки такі помилкові виявлення існують лише в результаті помилки в процесі виведення RCNN, яка, мабуть, не змогла локалізувати окремих людей.

У такому випадку, пошук виконується в зворотному порядку: спочатку розглядається фрагмент тексту запиту, а результатом є області зображень у всьому наборі тестів, що мають найвищий середній бал для кожного слова у запиті. Приклади таких запитів наведені на рисунку 3.9.

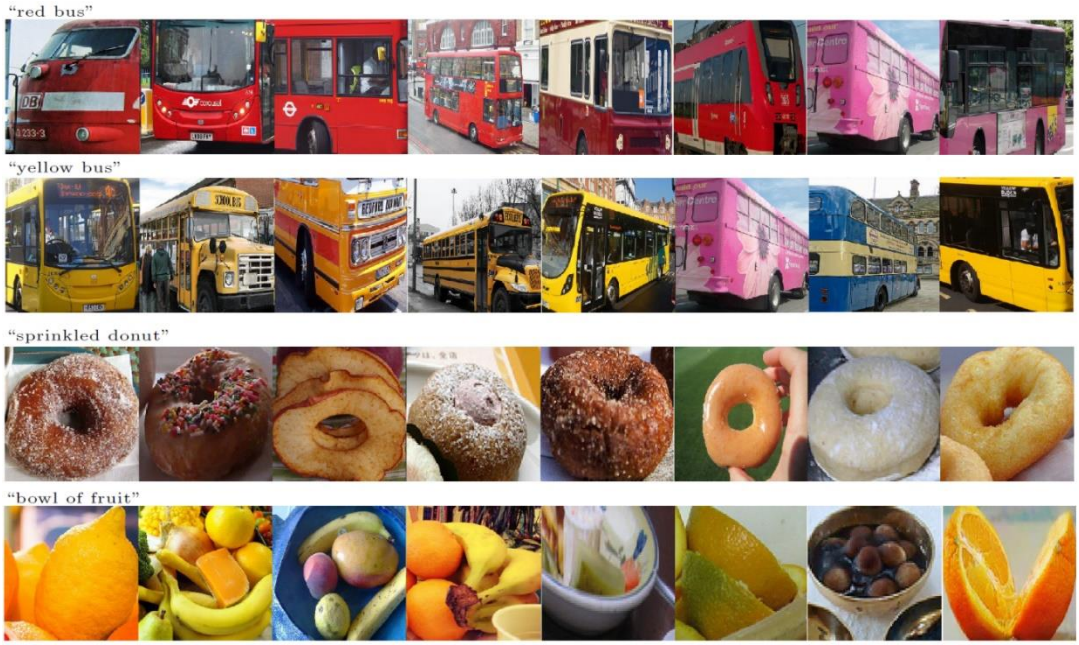

Рисунок 3.9 – Приклади областей з найбільшим балом

Метод чутливий до складних слів та модифікаторів. Наприклад, «червоний автобус» та «жовтий автобус» дають дуже різні результати. Крім того, якість результатів погіршується для менш рідкісних понять, таких як «тарілка з фруктами». Незважаючи на це, при маркуванні були

розглянуті всі візуальні види текстових фрагментів із необроблених даних повних зображень та речень, без явних відповідностей.

На рисунках 3.10 та 3.10 показано додаткові можливості методу – можливість моделювати важливість слів та регіонів шляхом масштабування величини їх вкладених векторів *i*, *d*.

Рисунок 3.10 – Тестові дані з Flickr30K встановлює регіони з великою векторною величиною

Magnitude	Word	Magnitude	Word
0.42	now	2.61	kayaking
0.42	simply	2.59	trampoline
0.43	actually	2.59	pumpkins
0.44	but	2.58	windsurfing
0.44	neither	2.56	wakeboard
0.45	then	2.54	acrobatics
0.45	still	2.54	sousaphone
0.46	obviously	2.54	skydivers
0.47	that	2.52	wakeboarders
0.47	which	2.52	skateboard

Рисунок 3.11 –Найвищі та найнижчі значення показників відповідності слова та фрагмента зображення

Відповідно до рисунків 3.10 та 3.11 , можна спостерігати, що представлення візуально виразних слів, таких як «байдарки, гарбузи», як правило, мають більшу величину в просторі вкладання, що означає більший вплив на кінцеві показники зображення-речення завдяки внутрішньому продукту. І навпаки, модель вчиться наносити на зупинку такі слова, як «зараз, просто, насправді, але» біля початку, що зменшує їх вплив.

ВИСНОВКИ

В кваліфікаційній роботі магістра вирішено задачу удосконалення гібридної згорткової нейронної мережі, яка призначена для маркування візуальних даних фрагментами тексту природною мовою.

При виконанні магістерської кваліфікаційної роботи було проаналізовано властивості аналіз характеристик когнітивних систем та сервісів та показано, що такі сервіси широко використовуються у фоновому режимі для підтримки взаємодії людини з корпоративними сайтами багатьох сучасних фірм, зокрема IBM та Microsoft.

При досліджені задачі комп’ютерного зору та проаналізовано підходи до розпізнавання образів, а також методи обробки природних мов. За результатами аналізу показано важливість поєднання зображень та тексту, коли для кожного зображення задається речення, що описує це зображення.

Для методу маркування зображень текстом в кваліфікаційній роботі обґрунтовано використання критерію оцінки сумісності зображень та тексту. Цей критерій враховує кути між двома векторами – зображень та тексту. Критерій використовується при співставленні двох векторів. Перший вектор кодує послідовність зображень, а другий – послідовність речень. Метод ранжує текст для кожного зображення.

Проведена експериментальна перевірка показала, що при ранжуванні слів для зображень більш загальні слова мають менше значення рейтингової оцінки. Слова, які більш детально описують зображення, мають більше значення оцінки. Таким чином, зображення маркується текстом, що дає більш точний опис зображення.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Bermúdez J.L. Cognitive Science: An Introduction to the Science of the Mind, Cambridge University Press; 2nd edition. P. 552.

2. Schalkoff R.J. Artificial Neural Networks. NY: The McGraw Hill Comp., Inc., 1997. 448 p.

3. Agarwal S., Roth D. Learning a sparse representation for object detection. Springer, Heidelberg: ECCV 2002. LNCS, 2002, 2353. P. 113–130.

4. Agre P., Chapman D. Pengi: An implementation of a theory of situated action, 1987.

5. Allen J., Byron D., Dzikovska M., Ferguson G., Galescu L. An architecture for a generic dialogue shell. Journal of Natural Language Engineering, 2000, 6(3). P. 1–16.

6. Ahuja N., Schachter B. Image models, ACM Computing Surveys 1983, 15(1). P. 83– 84.

7. Garding J. Shape from texture for smooth curved surfaces. Proc. European Conference on Computer Vision (ECCV), 1992. P. 630–638.

8. Бодянский Е.В., Руденко О.Г. Искусственные нейронные сети: архитектуры, обучение, применение. Харьков: ТЕЛЕТЕХ, 2004. 372 с.

9. Осовский С. Нейронные сети для обработки информации/Пер. с польского И.Д. Рудинского. М.: Финансы и статистика, 2002. 344 с.

10. Abe N., Warmuth M. K. On the computational complexity of approximating distributions by probabilistic automata. Machine Learning, 1992. P. 205–60.

11. Fraser A., Marcu D. Semi-supervised training for statistical word alignment. Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, 2006. P. 769–76.

12. Abney, S. A computational model of human parsing. Journal of Psycholinguistic Research, 1989. P.129–44.

13. Jinying CH., Palmer M. Improving English verb sense disambiguation performance with linguistically motivated features and clear sense distinction boundaries. Language Resources and Evaluation, 2009. P. 181–208.

14. Abney, S. Statistical methods and linguistics. The Balancing Act: Combining Symbolic and Statistical Approaches to Language. Cambridge, MA: MIT Press, 1996. P. 2–26.

15. Bengio Y., Ducharme R., Vincent P., Janvin Chr. A neural probabilistic language model. The Journal of Machine Learning Research, 2003. P. 1137–1155.

16. Sutskever I., Martens J., Hinton G.Generating text with recurrent neural networks. International Conference on Machine Learning, 2011.

17. Mikolov T., Karafiat M., Burget L., Cernockiy Y., Khudanpur S. Recurrent neural network based language model. In INTERSPEECH, 2010.

18. LeCun Y., Bottou L., Bengio Y., Haffner P.Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11). P. 2278–2324.

19. Russakovsky O., Deng J., Su H., Krause J., Satheesh S. Karpathy A., Fei- Fei L. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 2015, 115(3). P.211–252.

20. Krizhevsky A., Sutskever I., Hinton G. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems, 2012.

21. Deng J., Dong W., Socher R., Li-Jia L., Li K., Fei-Fei L.Imagenet: A large-scale hierarchical image database. IEEE conference on Computer Vision and Pattern Recognition, 2009.

22. Mikolov T., Sutskever I., Chen K., Corrado G., Dean J. Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems, 2013.

23. Socher R., Karpathy A., V Le Q., Manning Chr., Y Ng A. Grounded compositional semantics for finding and describing images with sentences. TAssociation for Computational Linguistics, 2014.

24. Karpathy A., Joulin A., Fei-Fei L. Deep fragment embeddings for bidirectional image sentence mapping. In Advances in neural information processing systems, 2014.

25. Hodosh M., Young P., Hockenmaier J.Framing image description as a ranking task: data, models and evaluation metrics. Journal of Artificial Intelligence Research, 2013.

26. Young P., Lai A., Hodosh M., Hockenmaier J. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. TAssociation for Computational Linguistics, 2014.

27. Quoc V Le. Building high-level features using large scale unsupervised learning. 2013 IEEE international conference on acoustics, speech and signal processing, 2013.,P. 8595–8598.

28. Kiros R., Salakhutdinov R., Zemel R.Unifying visual-semantic embeddings with multimodal neural language models. Transactions of the Association for Computational Linguistics, 2015.

29. Vinyals O., Toshev A., Bengio S., Erhan D. Show and tell: A neural image caption generator. IEEE Conference on Computer Vision and Pattern Recognition, 2015, P. 3156–3164.

30. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations, 2015.

31. Szegedy C., Liu W., Jia Y., Sermanet P., Reed S. Going deeper with convolutions. IEEE Conference on Computer Vision and Pattern Recognition, 2015.