

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

«На правах рукопису»
УДК 004.89

До захисту допущено

В. о. завідувача кафедри
ММСА

_____ О.Л.Тимощук

«___»_____2018 р.

Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз
на тему: «Система розпізнавання тексту пошкоджених рукописних історичних
документів із застосуванням нейронних мереж»

Виконала:

студентка II курсу, групи КА-71 мп
Мулява Ольга Ярославівна _____

Керівник: в. о. завідувача кафедри ММСА
кандидат технічних наук, доцент, Тимощук О. Л. _____

Рецензент: завідувач кафедри загально-інженерних дисциплін та теплоенергетики
Таврійського національного університету
доктор технічних наук, професор
Медведєв М. Г. _____

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студентка _____

Київ

2018

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)

Спеціальність (спеціалізація) — 124 «Системний аналіз» («Системний аналіз і управління»)

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри ММСА

О. Л. Тимошук

«___» _____ 2018 р.

ЗАВДАННЯ

на магістерську дисертацію студентці Муляві Ользі Ярославівні

1. Тема дисертації: «Система розпізнавання тексту пошкоджених рукописних історичних документів із застосуванням нейронних мереж», науковий керівник дисертації Тимошук Оксана Леонідівна, кандидат технічних наук, доцент, затверджені наказом по університету від «07» листопада 2018 р. № 4121-с

2. Термін подання студентом дисертації: _____

3. Об'єкт дослідження: фото історичних пошкоджених документів

4. Предмет дослідження: методи обробки і аналізу зображень

5. Перелік завдань, які потрібно розробити:

- 1) Огляд технічної літератури за темою роботи;
- 2) Дослідження актуальності обраної теми;
- 3) Вибір методів для моделювання і прогнозування;
- 4) Збір вхідних даних;
- 5) Виконання обчислювальних експериментів;
- 6) Аналіз результатів моделювання і прогнозування;
- 7) Проведення аналізу ринкових можливостей запуску стартап-проекту;
- 8) Підготовка ілюстративного матеріалу;
- 9) Оформлення пояснювальної записки.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу:

- 1) Постановка завдання дослідження;
- 2) Методи байєсівського аналізу даних;
- 3) Співвідношення гранулярного фільтра;
- 4) Наукова новизна результатів.

7. Орієнтовний перелік публікацій:

(1) Система розпізнавання тексту пошкоджених рукописних історичних документів із застосуванням нейронних мереж // Системні дослідження та інформаційні технології. – Стаття подана в редакцію журналу.

8. Дата видачі завдання: _____**Календарний план**

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Отримання завдання на магістерську дисертацію	07.09.2018 – 09.09.2018	
2.	Огляд технічної літератури за темою	10.09.2018 – 30.09.2018	
3.	Дослідження актуальності обраної теми	01.10.2018 – 07.10.2018	
4.	Вибір методів для моделювання і прогнозування	08.10.2018 – 14.10.2018	
5.	Збір вхідних даних	15.10.2018 – 21.10.2018	
6.	Виконання обчислювальних експериментів	22.10.2018 – 28.10.2018	
7.	Аналіз результатів моделювання і прогнозування	29.10.2018 – 04.11.2018	
8.	Проведення аналізу ринкових можливостей запуску стартап-проекту	05.11.2018 – 11.11.2018	
9.	Підготовка ілюстративного матеріалу	12.11.2018 – 18.11.2018	
10.	Оформлення пояснювальної записки	19.11.2018 – 26.11.2018	

Студентка

О. Я. Мулява

Науковий керівник дисертації

О. Л. Тимощук

РЕФЕРАТ

Магістерська дисертація: 98 с., 15 рис., 20 табл., 18 джерел, 1 додаток

Метою дослідження є дослідження і реалізація розпізнавання рукописного історичного тексту за допомогою кроків первинної обробки і нейронних мереж, а також розробка програмного продукту, який розпізнає стародавній іврит із фото пошкоджених рукописів.

Об'єктом дослідження є фото історичних пошкоджених документів, написаних вручну на стародавньому івриті, які в кінцевому результаті мають бути розпізнані як текст.

Предмет дослідження є методи первинної обробки зображень такі, як бінаризація та алгоритми визначення контурів на зображенні, сегментація ліній тексту методом гісторгами, згорткова нейронна мережа, підхід ковзного вікна для розпізнавання символів.

Методи дослідження ґрунтуються на машинному навчанні, методах оптимізації, математичній статистиці.

Актуальність теми - робота присвячена дослідженню розпізнавання рукописного тексту як цінного історичного джерела за допомогою передових технологій у сфері інформаційних технологій.

РОЗПІЗНАВАННЯ РУКОПИСНОГО ТЕКСТУ, АНАЛІЗ ЗОБРАЖЕНЬ, МАШИННЕ НАВЧАННЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА.

ABSTRACT

The theme: 'Handwriting recognition system for degraded historic manuscripts using neural networks.'

Diploma work: 98 p., 15 fig., 20 tabl., 1 appendixes, 18 references.

The purpose of the study is to research and realize the recognition of handwritten historical text through the steps of primary processing and neural networks, as well as the development of a software product that recognizes ancient Hebrew from photographs of damaged manuscripts.

The object of the study is photos of historical damaged documents written manually in ancient Hebrew, which should be recognized as text.

The subject of research is the methods of image pre-processing such as binarization and algorithms for determining the contours in the image, segmentation of the text lines by the high torque method, and the convolutional neural network.

Methods based on machine learning, optimization methods, mathematical statistics.

The relevance of the study - the work is devoted to the study of recognition of handwritten text as a valuable historical source with the help of advanced technologies in the field of information technology

HANDWRITING RECOGNITION, IMAGE PROCESSING, MACHINE LEARNING, ARTIFICIAL INTELLIGENCE, CONVOLUTIONAL NEURAL NETWORK.

ЗМІСТ

ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	8
ВСТУП.....	9
РОЗДІЛ 1 РОЗПІЗНАВАННЯ ТЕКСТУ У КОНТЕКСТІ ШТУЧНОГО ІНТЕЛЕКТУ	11
1.1 ШТУЧНИЙ ІНТЕЛЕКТ	11
1.2 РОЗПІЗНАВАННЯ ТЕКСТУ ЗА ДОПОМОГОЮ МАШИННОГО НАВЧАННЯ..	15
1.3 ІНСТРУМЕНТИ І ТЕХНОЛОГІЇ МАШИННЕ НАВЧАННЯ	22
1.4 ВИСНОВКИ ДО РОЗДІЛУ	24
РОЗДІЛ 2 МЕТОДИ ОБРОБКИ І РОЗПІЗНАВАННЯ ТЕКСТУ	26
2.1 МЕТОДИ ПЕРВИННОЇ ОБРОБКИ ЗОБРАЖЕНЬ.....	26
2.2 СЕГМЕНТАЦІЯ ЛІНІЙ ТЕКСТУ	35
2.3 ОПРАЦЮВАННА ШУМУ НА ЗОБРАЖЕННЯХ.....	37
2.3 ЗГОРТКОВА НЕЙРОННА МЕРЕЖА	41
2.4 ВИСНОВКИ ДО РОЗДІЛУ	43
РОЗДІЛ 3 ЗАПРОПОНОВАНА СИСТЕМА ОБРОБКИ РУКОПИСНОГО ТЕКСТУ	44
3.1 БІНАРИЗАЦІЯ ВИХІДНОГО ЗОБРАЖЕННЯ	44
3.2 ЗНАХОДЖЕННЯ ТА ПОПЕРЕДНЯ ОБРОБКА ТЕКСТОВОЇ ОБЛАСТІ	46
3.3 СЕГМЕНТАЦІЯ ЛІНІЙ ТЕКСТУ	48
3.4 РЕКОНСТРУКЦІЯ СИМВОЛІВ ТА ОПРАЦЮВАННЯ ШУМУ	49
3.5 МОДЕЛЬ-КЛАСИФІКАТОР ЛІТЕР.....	49
3.6 ЗБАГАЧЕННЯ ДАНИХ.....	52
3.8 ОЦІНКА МОДЕЛІ	54
3.9 ВИБІР ПРОГРАМНОЇ ПЛАТФОРМИ	56
3.10 ВИСНОВКИ ДО РОЗДІЛУ	61

РОЗДІЛ 4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ	62
4.1 ОПИС ІДЕЇ ПРОЕКТУ	62
4.2 ТЕХНОЛОГІЧНИЙ АУДИТ ІДЕЇ ПРОЕКТУ	64
4.3 АНАЛІЗ РИНКОВИХ МОЖЛИВОСТЕЙ ЗАПУСКУ СТАРТАП-ПРОЕКТУ	65
4.4 РОЗРОБЛЕННЯ РИНКОВОЇ СТРАТЕГІЇ ПРОЕКТУ	75
4.5 РОЗРОБЛЕННЯ МАРКЕТИНГОВОЇ ПРОГРАМИ СТАРТАП-ПРОЕКТУ	81
4.6 ВИСНОВКИ ДО РОЗДІЛУ	86
ВИСНОВКИ	87
ПЕРЕЛІК ПОСИЛАНЬ	88
ДОДАТОК А ЛІСТИНГ ПРОГРАМИ.....	90

ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ШІ – штучний інтелект

ML (Machine Learning) – машинне навчання

CNN (Convolutional Neural Network) – згорткова нейронна мережа

ROC (Receiver Operating Characteristic) – робоча характеристика приймача

OCR (Optical Character Recognition) – оптичне розпізнавання символів

ВСТУП

Рукопис залишається засобом спілкування та збору інформації в повсякденному житті навіть із впровадженням нових технологій. Більш того, величезна кількість історичних колекцій не є доступна у зручному форматі. В даний час підходи та способи перетворення зображень в цифровий текст розвиваються швидкими темпами, навіть незважаючи на те, що все ще є можливості для покращення та вирішення багатьох завдань.

В останні роки системи розпізнавання друкованого тексту стали досить ефективними. Рукописні цифри та інтерактивне написання розпізнаються якісно. Однак поточні технології все ще знаходяться на обмеженому рівні, щоб розпізнавати текстові зображення різних стилів та мов рукописного вводу [1].

У цьому документі запропонований підхід системи розпізнавання рукописного тексту, починаючи з ідеї, що хороша попередня обробка створює основу для успішного розпізнавання. До визнання себе як попередньої обробки підхід включає в себе бінаризацію, пошук текстової області, подальшу обробку, такі як коригування фону, сегментацію ліній, реконструкцію символів та обробку шумів.

Системи розпізнавання рукописного тексту можуть відрізнитися в багатьох аспектах і залежать від конкретних завдань. Бінаризація може виконуватися як перший крок або після сегментованих рядків або символів. Наш підхід використовує його як перший крок, оскільки наступні кроки потребують бінарних зображень, і, крім того, він ефективний завдяки появі вхідних зображень. Потім знайдеться текстова область, і зображення буде налаштовано, так що є білий фон та чорні літери. Цей крок є досить значним, оскільки він впливає на етапі сегментації ліній. Слід зазначити, що проекція гістограми для сегментації ліній є звичайною практикою. У цьому документі цей підхід до сегментації ліній змінюється з способом пошуку піків у гістограмі. Слід

розглянути питання про деградовані історичні документи, реконструкцію персонажа. На цьому кроці запропонована система вирішує проблему шуму.

Після отримання текстових рядків виникає складність виконання точної сегментації характеру. Існує практика сегментації букв та класифікації букв окремо. Наприклад, використовуючи аналіз підключених компонентів або проектування гістограми лінії. Проте деградовані або перекриваючі символи можуть бути визнані неправильними. Наш підхід передбачає одночасне розпізнавання символів та сегментацію.

Ця робота організована наступним чином. У розділі 2 описані запропоновані методи та всі його кроки. У розділі 3 представлені та проаналізовані експериментальні результати. Нарешті, деякі дискусії та заключні зауваження наведені в розділі 4.

РОЗДІЛ 1 РОЗПІЗНАВАННЯ ТЕКСТУ У КОНТЕКСТІ ШТУЧНОГО ІНТЕЛЕКТУ

Інформаційні технології проникли у життя соціуму у всіх його сферах, допомагаючи людству у розвитку. Яскравим прикладом цього є розробка нової форми розуму, а саме, - штучного інтелекту.

1.1 Штучний інтелект

Штучний інтелект є галуззю інформатики, метою якої є створення інтелектуальних машин. Це стало важливою частиною технологічної галузі.

Дослідження, пов'язані з штучним інтелектом, дуже технічні та спеціалізовані. Основні проблеми штучного інтелекту включають в себе програмування комп'ютерів для певних рис, таких як: знання, міркування, вирішення проблем, сприйняття, навчання, планування, можливість маніпулювати та переміщати об'єкти.

Інжиніринг знань є основною частиною досліджень AI. Машини часто можуть діяти і реагувати, як людина, лише в тому випадку, якщо вони мають багату інформацію, що стосується світу. Штучний інтелект повинен мати доступ до об'єктів, категорій, властивостей та відносин між ними для здійснення знань. Ініціювання здорового глузду, міркування та вирішення проблем у машинах - складне і нудне завдання.

Машинне навчання також є основною частиною AI. Навчання без будь-якого нагляду вимагає здатності виявляти закономірності у потоках вхідних даних, тоді як навчання з адекватним наглядом передбачає класифікацію та чисельні регресії. Класифікація визначає категорію, до якої об'єкт належить, і операції регресії з отриманням набору числових вхідних або вихідних прикладів,

тим самим відкриваючи функції, що дозволяють створювати відповідні виходи з відповідних входів. Математичний аналіз алгоритмів машинного навчання та їх продуктивність є чітко визначеною гілкою теоретичної інформатики, яку часто називають теорією обчислювальної навчання.

Машинна сприйняття стосується здатності використовувати сенсорні входи для виведення різних аспектів світу, а комп'ютерне бачення - це сила для аналізу візуальних входів з кількома під-проблемами, такими як розпізнавання обличчя, об'єкта та жестів.

Робототехніка також є основним напрямком, пов'язаним з AI. Роботи вимагають інтелекту для вирішення таких завдань, як маніпулювання об'єктами та навігація, а також підпроблеми локалізації, планування руху та картографування.

Розробка штучного інтелекту як наряду наряду бере розвиток з наук як психологія, математика, нейрофізіологія, та інформаційні технології.

Термін "штучний інтелект" був створений у 1956 році, однак AI став більш популярним сьогодні завдяки збільшенню об'ємів даних, розширені алгоритми та покращенню обчислювальної потужності та зберігання.

Ранні дослідження AI у 1950-х роках вивчало такі теми, як вирішення проблем та символічні методи. У 1960-х роках Міністерство оборони США зацікавилось подібною роботою і почало навчання комп'ютерів, щоб імітувати базові людські міркування. Наприклад, в 70-х роках минулого століття в Агентстві передових дослідницьких проектів у галузі оборони (DARPA) було завершено проекти з планування вулиць. І DARPA виготовив інтелектуальних персональних помічників у 2003 році, задовго до того, як Сірі, Алекса або Кортана були домашніми іменами.

Ця рання робота проклала шлях до автоматизації та формальних міркувань, які ми бачимо на сучасних комп'ютерах, включаючи системи підтримки прийняття рішень та інтелектуальні пошукові системи, які можуть бути розроблені для доповнення та збільшення людських здібностей.

Хоча голлівудські фільми та науково-фантастичні романи зображують AI як людських роботів, які переймають світ, поточна еволюція технологій AI не така страшна - або зовсім така розумна. Замість цього AI розвивається, щоб забезпечувати багато конкретних переваг у кожній галузі. Продовжуйте читати сучасні приклади штучного інтелекту в галузі охорони здоров'я, роздрібної торгівлі тощо.

У найближчій перспективі мета збереження впливу AI на суспільство вигідне, мотивація досліджень у багатьох сферах, від економіки та права до технічних тем, таких як перевірка, обґрунтованість, безпека та контроль. У той час як це може бути трохи більше, ніж незначні незручності, якщо ваш ноутбук збій або зламаний, стає все важливішим, що система AI робить те, що ви хочете це зробити, якщо вона контролює вашу машину, ваш літак, ваш кардіостимулятор, вашу автоматичну торгівлю системи або ваша енергосистема. Ще одним короткостроковим завданням є запобігання спустошливої гонки озброєнь в умовах смертельної автономної зброї.

У довгостроковій перспективі важливим питанням є те, що станеться, якщо квест за сильним II досягне успіху, і система AI стане кращою за людьми на всіх пізнавальних завданнях. Як зазначив І.Я. Гарний у 1965 році, проектування більш розумних систем II є пізнавальним завданням. Така система може потенційно зазнати рекурсивного самовдосконалення, викликаючи вибух інтелекту, що залишає людський інтелект далеко позаду. Винаймаючи революційні нові технології, така суперінтелекція може допомогти нам усунути війну, хвороби та бідність, і тому створення сильного ІС може бути найбільшою подією в історії людства. Деякі експерти висловлюють занепокоєння, однак, що це також може бути останнім, якщо ми не навчимося узгоджувати цілі AI з нашими, перш ніж він стане суперінтелектуальним.

Є хтось, хто сумнівається, чи буде досягнутий сильний AI, та інші, які наполягають на тому, що створення суперінтелектуального AI гарантовано буде корисним. У FLI ми усвідомлюємо обидві ці можливості, але також визнаємо потенціал системи штучного інтелекту навмисно або ненавмисно завдати

великої шкоди. Ми вважаємо, що сьогоднішні дослідження допоможуть нам краще підготуватися до таких потенційно негативних наслідків і запобігати таким майбутнім, таким чином, користуючись перевагами ІС, уникаючи підводних каменів.

АІ автоматизує повторюване навчання та пошук за допомогою даних. Але АІ відрізняється від апаратної, автоматичної автоматизованої роботи. Замість автоматизації ручних завдань, АІ надійно та без втоми виконує часто, великомасштабні комп'ютеризовані завдання. Для цього типу автоматизації, людський запит, як і раніше, є необхідним для створення системи та запиту правильних питань.

АІ додає інтелект для існуючих продуктів. У більшості випадків АІ не буде продаватися як індивідуальна заявка. Швидше за все, продукти, які ви вже використовуєте, будуть покращені завдяки можливостям АІ, подібно до того, як додано Siri як функцію нового покоління продуктів Apple. Автоматизація, розмовні платформи, боти і інтелектуальні машини можуть поєднуватися з великою кількістю даних для вдосконалення багатьох технологій вдома та на робочому місці, від розвідки безпеки до інвестиційного аналізу.

АІ адаптується за допомогою алгоритмів прогресивного навчання, щоб дати дані програмуванню. А. І. знаходить структуру та закономірності в даних, так що алгоритм набуває вміння: алгоритм стає класифікатором або предиктором. Отже, так само, як алгоритм може навчити себе, як грати в шахи, він може навчити себе, який продукт рекомендувати в Інтернеті. І моделі адаптуються при наданні нових даних. Зворотне розповсюдження - це техніка АІ, яка дає змогу коригувати модель шляхом навчання та додавання даних, коли перша відповідь не зовсім правильна.

АІ аналізує більш глибокі дані, використовуючи нейронні мережі, що мають багато прихованих шарів. Створення системи виявлення шахрайства з п'ятьма прихованими шарами було майже неможливо кілька років тому. Все, що змінилося, з неймовірною потужністю комп'ютера та великими даними. Вам потрібно багато даних, щоб тренувати глибокі моделі навчання, оскільки вони

навчаються безпосередньо з даних. Чим більше даних ви можете годувати їх, тим точніше вони стануть.

AI досягає неймовірної точності через глибокі нейронні мережі - це було раніше неможливо. Наприклад, ваша взаємодія з Alexa, Google Search та Google Photos залежить від глибокого навчання, і вони постійно стають точнішими, тим більше ми їх використовуємо. У галузі медицини методика AI з глибокого вивчення, класифікації зображень та розпізнавання об'єктів тепер може бути використана для пошуку раку на МРТ з такою ж точністю, як висококваліфіковані радіологи.

AI отримує найбільшу кількість даних. Коли алгоритми самонавчаються, самі дані можуть стати інтелектуальною власністю. Відповіді в даних; ви просто повинні застосувати II, щоб їх вивести. Оскільки роль даних зараз важливіша, ніж будь-коли раніше, вона може створити конкурентну перевагу. Якщо у вас є найкращі дані у конкурентній галузі, навіть якщо всі застосовують подібні методи, найкращі дані переможуть.

1.2 Розпізнавання тексту за допомогою машинного навчання

Розпізнавання рукописного тексту стало активною та складною областю досліджень. Система розпізнавання рукописного тексту відіграє дуже важливу роль у сучасному світі. Розпізнавання рукописного тексту дуже популярне і обчислювально дорога робота. В даний час дуже важко знайти правильне значення рукописних документів. Є багато областей, де нам потрібно визнати слова, алфавіти та цифри. Є багато поштових адрес додатків, банківська перевірка, де нам потрібно визнати почерк. Цей оглядовий документ буде зосереджений на різній техніці, яка використовується при розпізнаванні рукописного тексту. Існує, в основному, два різних типи розпізнавання рукописного тексту в режимі онлайн та розпізнавання рукописного вручення. Для автономного режиму розпізнавання рукописного вводу існує багато

підходів. Цей оглядовий документ буде представляти обмеження та переваги різної техніки, яка використовується для розпізнавання рукописного тексту. Отже, розпізнавання рукописного тексту було вивчено з багатьох десятиліть. Система розпізнавання рукописного тексту може бути використана для вирішення багатьох складних проблем і може зробити роботу людини легкою.

Розпізнавання рукописного вводу - це здатність і техніку системи, які отримують вхід від сенсорного екрана, електронного пера, сканера, зображень та паперових документів. Система розпізнавання рукописного вводу в автономному режимі є мистецтвом ідентифікації слова з зображення. Як ми знаємо, кожна людина має свій різний стиль письма, тому дуже важко визнати правильні рукописні символи та цифри. Система розпізнавання рукописного тексту розроблена для досягнення точності та надійності роботи. Отже, розпізнавання рукописного тексту є найбільш складною областю, якщо розпізнавати зображення та образ. Розпізнавання рукописного тексту дуже корисно в реальному світі. Є багато практичних проблем, коли система розпізнавання рукописного тексту дуже корисна, як аналіз документації, інтерпретація поштової адреси, обробка банківських чеків, перевірка підпису, поштова адреса. Кілька підходів використовувались як в режимі розпізнавання рукописного вводу в режимі онлайн, так і в автономному режимі, як статистичні методи, структурні методи, нейронна мережа та синтаксичні методи. Деякі системи розпізнавання визначають штрихи, інші застосовують розпізнавання на одному символі або цілі слова. Отже, система розпізнавання рукописного тексту - це робота як середовище спілкування між людиною та машиною. На малюнку 1 наведено приклад деяких рукописних слів та цифр.

Система оптичного розпізнавання символів була вивчена протягом останніх десятиліть. У 1914 році Еммануель Голдберг розробив систему, яка читає рукописні символи та цифри і потім конвертується в телеграфний код. У той же час Едмунд Фурньє д'Альбе розробив сканер *Optofon*, який сканує друковану сторінку та випускає вихід. Голдберг продовжував розвивати систему розпізнавання рукописного введення даних. Через деякий час він запропонував

підібрати зображення з шаблонами, що містять ідентифікацію бажання. Цей метод відомий як спосіб відповідності шаблону.

Після цього Пауль У. Хандель також запропонував американський патент на технологію рукописного впорядкування шаблонів у США у 1933 році. У 1994 році інженери RCA запропонували спочатку первісний комп'ютерний тип оптичного розпізнавання символів для допомоги сліпим людям. Він призначений для перетворення рукописного звіту в перфокарти для введення в комп'ютер для допомоги у обробці відвантаження 20-25 мільйонів книг на рік. У 1965 році Reader's Digest та RCA співпрацювали з метою створення оптичної системи розпізнавання символів. У 1985 році запропоновано структурні підходи зі статистичними методами. У цих системах символи розбиті на безліч моделей, таких як горизонтальні та вертикальні лінії та різні криві. У цьому методі система зосереджена на формі персонажів. Після 1990 року реальний прогрес досягнуто за допомогою нових методів та методологій обробки зображень та розпізнавання образів. У сучасному світі використовуються більш потужні комп'ютери та більш точні пристрої, такі як електронна ручка, сканер та планшети. Багато підходів, таких як НММ, нейронна мережа, алгоритм зворотного поширення, нечітка нейронна мережа, використовуються для розпізнавання рукописних документів.

Розпізнавання рукописного персонажа є одним з практично важливих проблем у програмах розпізнавання образів. Програми розпізнавання розрядів включають в себе сортування поштових відправлень, обробку банківських чеків, введення даних у формі тощо. Основним завданням є здатність розробляти ефективний алгоритм, здатний розпізнавати цифри, написані рукописним словом, і який способом подає користувачі сканера, планшета та інших цифрових пристроїв.

Незважаючи на велику кількість інструментів для написання технологічних текстів, багато людей все-таки вирішують брати свої нотатки традиційно: з ручкою та папером. Проте є недоліки в написанні тексту.

Ефективно зберігати і отримувати доступ до фізичних документів важко, ефективно їх шукати та ділитися ними з іншими.

Таким чином, багато важливих знань втрачаються або не переглядаються через те, що документи ніколи не передаються в цифровий формат. Таким чином, ми вирішили вирішити цю проблему у нашому проекті, оскільки ми вважаємо, що значно більша зручність керування цифровим текстом порівняно з письмовим текстом допоможе людям більш ефективно отримувати доступ, шукати, ділитися ними та аналізувати їхні записи, досі дозволяючи використовувати їх бажаний метод письма.

Виявлення рукописного тексту - це техніка або здатність комп'ютера отримувати та інтерпретувати зрозумілі рукописні введення з джерела, наприклад паперові документи, сенсорний екран, фотографії тощо. Коли ми щось пишемо і подаємо як вхід до системи, то як ми розпізнаємо ці слова та рукописні документи. Цей процес виконується системою розпізнавання рукописного тексту. Отже, можна сказати, що система розпізнавання рукописного тексту може бути використана як паралельний термін. Симуляція машини людського письма - це область застосування системи розпізнавання рукописного тексту. Система розпізнавання рукописного тексту є способом інтерфейсу між людиною та машиною. Є n спосіб написати таке слово, як слово може бути вписаним курсором, літеру верхнього і нижнього регістрів. Отже, методи розпізнавання рукописного тексту мають можливість розпізнавати текст курсору і т. Д.

З кількох десятиліть система розпізнавання рукописного тексту є найбільш розвиненою дослідницькою сферою. Тому що сьогодні потреба системи розпізнавання рукописного тексту зростає з кожним днем. Тому що написання - це природний спосіб зберігання інформації та передачі інформації. Рукопис не забезпечує спосіб спілкування між людиною і людиною, він також забезпечує інтерфейс між людиною та машиною.

Розпізнавання рукописного вводу - це здатність комп'ютера або мобільного пристрою читати рукопис як фактичний текст. Найпоширенішим

випадком використання в сучасному мобільному світі є розпізнавання рукописного тексту як прямий вхід до сенсорного екрану за допомогою стилуса або пальця. Це корисно, оскільки це дозволяє користувачеві швидко вписати числа та імена контактів у порівнянні з введенням тієї ж інформації за допомогою екранної клавіатури. Це пояснюється тим, що більшість людей більш комфортно писати і може зробити це швидко. Ця функція може бути непридатною для більшості смартфонів або планшетів, але доступно багато програм для розпізнавання рукописного вводу.

Оптичний розпізнавання символів (OCR) - це найбільш основний метод, який використовується для розпізнавання рукописного тексту. Це робиться шляхом сканування рукописного документа, а потім перетворення його в основний текстовий документ. Це також працює, якщо взяти зображення рукописного тексту. OCR - це, в основному, форма розпізнавання зображень, яка призначена для розпізнавання рукописного тексту замість обличчя або форми, наприклад орієнтирів.

З деякими недавніми завидними проривами у цій галузі, можливо, ви чули про OCR. Оптичний розпізнавання символів (OCR) став цікавою темою протягом багатьох років. Він визначається як процес оцифрування зображення документа в складові символи. Уявіть собі систему, яка може розшифрувати рецепт лікаря у поганому почерку, як зазначено вище, у нижній частині двох зображень. Незважаючи на десятиліття інтенсивних досліджень, розробка OCR з можливостями, порівнянними з можливостями людини, все ще залишається відкритим викликом.

Коли ви заповнюєте CAPTCHA, ви знаєте, що це працює над вашою здатністю перевершити завдання, яке вважалося недоступним для комп'ютерів. Якщо ви вважаєте, що CAPTCHA буде залишатися важко розбити кілька років, дозвольте нам сказати вам, ви не могли б помилятися.

За останні кілька років кількість академічних лабораторій та компаній, що займаються дослідженнями розпізнавання символів, різко зросла. OCR є складною проблемою через різноманітність мов, шрифтів і стилів, в яких текст

може бути написаний, і не кажучи вже про складні правила мов. Отже, методи різних галузей інформатики, тобто обробка зображень, класифікація моделей та обробка природної мови тощо, використовуються для вирішення різних завдань. Перш ніж ми почнемо вирішувати виклики у цій галузі, і чим займається група Хурана, давайте коротко поговоримо про застосування OCR.

OCR дозволяє створювати велику кількість корисних додатків в режимі онлайн і офлайн, таких як:

1. Введення даних для бізнес-документів. Уявіть, що хіміки / аптеки зможуть сканувати рукописів лікарів без будь-якої боротьби.
2. Автоматичне розпізнавання номерного знака. Ви знаєте, наскільки легко це призведе до впровадження правил дорожнього руху.
3. Автоматичні страхові документи, які можуть витягувати ключову інформацію без втручання людини.
4. Витяг інформації візитної картки в список контактів.
5. Сканування книг
6. Зробити електронні зображення друкованих документів для пошуку, наприклад, Google Books
7. Перетворення рукописного тексту в режимі реального часу для керування комп'ютером (ручне обчислення).
8. Допоміжні технології для сліпих та людей з вадами зору. Уявіть, що всі дорожні знаки доступні для сліпого на відстані.

Програми практично у всіх сферах життя. Ви просто повинні думати, де ви зустрінете письмовий текст, і ви знайдете заявку.

Проблема розпізнавання тексту була спробована багатьма різними підходами. Існує декілька способів нагрівання кота, і оскільки ОРС має великі гроші, люди намагаються багато.

Збіг шаблонів - це один із найпростіших і найстаріших підходів. У цьому багато шаблонів кожного слова зберігаються для вхідного зображення, обчислюється помилка або різниця з кожним шаблоном. Виводиться символ, що відповідає мінімальній похибці. Ця техніка ефективно працює для розпізнавання

стандартних шрифтів, але призводить до слабкої роботи з написаними вручну символами.

Матрична відповідність передбачає порівняння зображення зі збереженим гліфом на основі пікселів на пікселі; він також відомий як "відповідність шаблону", "розпізнавання образів" або "співвідношення зображень". Це покладається на те, що вхідний гліф правильно ізольований від решти зображення, а на збереженому гліфі знаходиться такий же шрифт і в тому ж масштабі. Ця техніка найкраще працює з машинописним текстом і не працює належним чином, коли виникають нові шрифти. Ця методика реалізує ранній фізичний орієнтований на основі фотоелементів.

Розпізнавання зображень - це ще один підхід, в якому аналізується статистичний розподіл точок та витягуються ортогональні властивості. Для кожного символу вектор функції розраховується та зберігається в базі даних. Причому розпізнавання виконується шляхом знаходження відстані вектора вектора вхідного зображення до зображення, збереженого в базі даних, і виведення символу з мінімальним відхиленням.

З геометричного підходу, з іншого боку, функції залежать від фізичних властивостей, таких як кількість стиків, відносних положень, кількості кінцевих точок, співвідношення довжини до ширини тощо. Класи, вимушені на основі цих геометричних ознак, досить відрізняються від незначного перекриття. Однак головним недоліком такого підходу є те, що цей підхід значною мірою залежить від набору символів, хіндійських алфавітів або англійських алфавітів або цифр тощо. Можливості, витягнуті для одного набору, навряд чи працюють для іншого набору символів. Ці фактори впливають на рівень розпізнавання та багато інших показників продуктивності в знак визнання персонажів.

1.3 Інструменти і технології машинного навчання

Застосування популярних алгоритмів машинного навчання для великих обсягів даних спричинили нові практичні проблеми. Традиційні бібліотеки машинного навчання не підтримують обчислення великі масиви даних, тому необхідні нові підходи.

Як наслідок, парадигма хмарних обчислень вплинули на сферу машинного навчання. Один з підходів – це використання статистичних інструментів та бібліотек (система R, Пайтон), застосованих в хмарі. Інший напрям продуктів – додавання існуючих інструментів з плагінами, які дозволяють користувачам створити кластер Хадуп в хмарі. Далі в списку бібліотеки розподіленої реалізації для алгоритмів машинного навчання, а також на передумові розгортання складних систем для аналізу даних і інтелектуальний аналіз даних. Останній підхід цього дослідження машинного навчання, програмне забезпечення як послуга (Software-as-a-Service), декілька стартапів великих даних (а також великі компанії) вже відкривають свої рішення на ринок[4].

Протягом більше двох десятиліть, продукти, які розробляються паралельно з базами даних, таких як Teradata, Oracle або Netezza надали кошти для реалізації алгоритмів машинного навчання в коді SQL, що є складним завданням для підтримки. Крім того, великомасштабні установки цих продуктів є дорогими і не є доступним варіантом у більшості випадків. Інший рушій для зсуву парадигми від реляційної моделі до інших альтернатив - це нова природа даних. Приблизно п'ять років тому, велика частина даних мала транзакційний характер, що складається з числових або рядкових даних, які легко зберігаються в рядках і стовпцях реляційної бази даних. З тих пір, у той час як структуровані дані мають лінійне зростання, неструктурованих (наприклад, аудіо та відео) і напівструктуровані дані (наприклад, дані веб-трафіку, соціальний медіа-контенту і т.д.) демонструють експоненціальне зростання. Велика частина нових

даних або напівструктурованого формату, тобто складається з заголовків, за якими слідує рядки тексту або чисті неструктуровані дані (фото, відео, аудіо). У той час як останні мають обмежений текстовий зміст і є більш складним для синтаксичного аналізу, напівструктуровані дані викликали безліч нереляційних сховищ даних (NoSQL), адаптованих для обробки величезної кількості даних. Як наслідок, за останні 5 років дослідники бачили перехід до паралелізації машинного навчання з використанням цих нових платформ, таких як NoSQL сховищ даних, розподілених середовищах обробки (MapReduce), або хмарних обчислень. Варто знадати гарну метафору Бена Уертера [18], співзасновник Platfora, для обробки великих даних: «Якщо говорити у термінах промислової революції, ми в доіндустріальній епосі ремісництва, що передує масовому виробництву. Це еквівалентно необхідності залучати експерта коваля викувати вилки і ложки для нашого обіднього столу» [5].

Машинне навчання за своєю суттю є трудомістким завданням, тобто багато зусиль було проведено з метою зменшення часу виконання. Парадигма хмарних обчислень і провайдери хмарних технологій виявилися цінними елементами для прискорення платформ машинного навчання. Таким чином, популярні статистичні інструменти середовища, як R, Octave, Python - перемістились в хмару також. Є два основних напрямки для їх інтеграції з постачальниками хмарних технологій: створення кластера в хмарі з самозавантаженням його статистичних інструментів, або розширення статистичних середовищ за допомогою плагінів, які дозволяють користувачам створювати кластери Hadoop у хмарі і запуснути роботу на них.

Середовища R, Octave, Maple та аналогічні пропозиції інфраструктур низького рівня для аналізу даних, які можуть бути застосовані для великих наборів даних, вже забезпечують постачальники хмарних технологій. Машинне навчання є те, що приходить на вершині цього і полегшує витяг корисних знань з великих обсягів даних для клієнтів, які мають слабкі знання у статистиці або взагалі не мають таких знань, тепер мають можливість автоматично отримувати так звані моделі знань з великих наборів даних. Спостерігається вибух стартапів

протягом останніх 5 років, деякі з них ще в прихованому режимі, які пропонують послуги машинного навчання своїм клієнтам і послуги аналізу великих даних. Ці ініціативи можуть бути PaaS / SaaS платформи або продукти, які можуть бути розгорнуті на приватних середовищах.

Огляд літератури і ринку, можна зробити висновок, що машинне навчання має різноманітні прояви. Можна класифікувати 5 підходів:

1. Середовище машинного навчання в хмарі (створити комп'ютерний кластер в хмарі і розгорнувши його інструментами статистики).
2. Модулі для інструментів машинного навчання - доповнення статистичних інструментів за допомогою плагінів, які дозволяють користувачам створювати кластер Hadoop в хмарі і виконувати завдання машинного навчання в ньому.
3. Розподілені бібліотеки машинного навчання - колекції паралелізованих реалізацій алгоритмів машинного навчання для розподілених середовищ (Hadoop, Dryad і т.д.).
4. Комплексні системи машинного навчання - продукти, які повинні бути встановлені в приватних центрах обробки даних (або в хмарі) і пропонують інтелектуальний аналіз даних високої продуктивності).
5. Постачальники програмне забезпечення як послуг для машинного навчання - PaaS / SaaS рішень, які дозволяють клієнтам отримати доступ до алгоритмів машинного навчання через веб-сервіси).

1.4 Висновки до розділу

З огляду на величезне нагромадження зібраних і доступних даних різних компаній, у промисловості, науці, техніці, аналіз таких даних стає все більш важливими. Сьогодні, дані, щоб бути проаналізованими, більше не обмежуються примітивним виглядом і класичними базами даних, а все більше і більше можуть включати в себе текстові документи і веб-сторінки (інтелектуальний аналіз

тексту, веб-сторінок та інше), просторові дані, мультимедійні дані, реляційні дані (соціальні мережі). Аналітичні інструменти дозволяють кінцевим користувачам збирати значущі зразки, приховані в великих обсягах структурованих і неструктурованих даних. Аналіз великих масивів даних дає користувачам можливість визначити нові джерела надходження, розвивати лояльні і рентабельні зв'язки з клієнтами, і вести загальну організацію більш ефективно, а також ефективно відносно вартості.

Дослідження в області виявлення знань і машинного навчання поєднує в собі класичні питання інформатики (алгоритми оптимізації, системи програмного забезпечення, баз даних) з елементами штучного інтелекту і статистики, аж до питань, орієнтованих на користувачів (візуалізація, інтерактивний інтелектуальний аналіз).

РОЗДІЛ 2 МЕТОДИ ОБРОБКИ І РОЗПІЗНАВАННЯ ТЕКСТУ

У методології машинного навчання важливим кроком є побудова математичних моделей. Ці моделі можуть бути отримані методами лінійної регресії, логістичної регресії, дискримінантного аналізу, дерев рішень, нейронних мереж і ін. Однак згорткова нейронна мережа найбільш часто використовується на практиці для аналізу зображень.

2.1 Методи первинної обробки зображень

Документальне зображення binarization це сегментація документа на передній план тексту та фону. Це зроблено для отримання чітких зображень, з яких текст можна легко отримати. Порогове значення використовується для сегментації зображень документа.

Документальне зображення binarization - це етап попередньої обробки для аналізу та обробки зображення документів. Це покращує ефективність технологій обробки документів, таких як OCR та аналіз макета. Image Binarization - це перетворення зображення документа на подвійне зображення документа. Пікселі зображення розділені на подвійний набір пікселів, тобто чорно-білий. Основною метою бінаризації зображення є сегментація документа на передній план тексту та фону.

Найпростіший підхід до бінаризації - пороговий. При пороговому значенні вибирається оптимальне порогове значення, і пікселі класифікуються як передній план або фон у порівнянні з цим пороговим значенням. Але в реальному житті деякі документи страждають від деградації, як нерівномірне висвітлення, фоновий шум, кровотеча та різноманітність освітленості та контрасту. Вибір

оптимального порогового значення для таких документів є складним завданням [1]. Неправильна оцінка порогового значення призводить до неправильної класифікації пікселів на передньому плані або фоні [2]. Це впливає на результати бінаризації та точності прикладів розпізнавання образів.

Як правило, бінаризація класифікується як глобальна та локальна [3]. Глобальна бінаризація - це техніка, в якій застосовується єдине порогове значення для подвійного зображення всього зображення. Це швидкий процес, але не працює для документів зі складним тлом. В Методі місцевої бінаризації замість одиничного порогу на ціле зображення, для кожного пікселя вибирається інше значення порогу. Поріг вибирається в залежності від пікселів сусідства. Місцеві пороги не дають хороших результатів для документів, що страждають фоновістю.

Методи бінаризації також можна класифікувати на основі критеріїв для вибору порогового значення. Деякі з методів вибору порога базуються на гістограмі, кластеризації, ентропії, локальних адаптивних методах.

Метод Оцу - це глобальний метод порогу, який перетворює зображення сірого масштабу на подвійний рівень зображення. Цей метод розбиває пікселі на два класи: один - передній план, а інший - фон. Він вибирає оптимальний поріг, який розділяє зображення на два різних класу. Порогова величина вибирається таким чином, що дисперсія всередині класу зводиться до мінімуму та дисперсія між класами максимізується. Метод Оцу дає найкращі результати лише для тих зображень, які мають чіткий бімодальний шаблон. Але деградовані документи, як правило, не мають такої чіткої схеми. Крім того, він не працює добре для зображень з нерівномірним підсвічуванням і тіні.

Логістична - популярна модель, яка використовується в аналізі виживання, яка може бути використана для оцінки значущості різних коваріат на тривалість життя людей або об'єктів за допомогою функції ризику. Крім того, може бути описаний кількісний вплив цих змінних протягом життя на важливі змінні результату (наприклад, медіани виживання).

Логістична регресія (англ. logistic regression) або логіт регресія (англ. logit model) — статистичний регресійний метод, що використовується у випадку коли залежна змінна може набувати тільки двох значень (чи, більш загально, скінченну множину значень).

Логістична регресія - окремий випадок узагальненої лінійної регресії. Припускається, що залежна змінна приймає два значення і має біноміальний розподіл:

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_k x_k \quad (2.1)$$

Її можна використовувати для завдання оцінки ймовірності результату події? Так, можна, обчисливши стандартні коефіцієнти регресії. Наприклад, якщо розглядається результат по позиці, задається змінна u зі значеннями 1 і 0, де 1 означає, що відповідний позичальник розплатився за кредитом, а 0, що мав місце дефолт. Однак тут виникає проблема: множинна регресія не «знає», що змінна відгуку бінарних за своєю природою. Це неминуче призведе до моделі з пророкує значеннями великими 1 і меншими 0. Але такі значення взагалі не припустимі для початкової задачі. Таким чином, множинна регресія просто ігнорує обмеження на діапазон значень для u .

Для вирішення проблеми завдання регресії може бути сформульована інакше: замість передбачення бінарної змінної, ми передбачаємо безперервну змінну зі значеннями на відріжку $[0,1]$ при будь-яких значеннях незалежних змінних. Це досягається застосуванням наступного регресійного рівняння (логіт-еквіваленті):

$$f(z) = \frac{1}{1 + e^{-z}} \quad (2.2)$$

Зробимо деякі перетворення:

$$z = \ln \frac{f(z)}{1-f(z)} = \ln \frac{p}{1-p} = \text{logit}(p) \quad (2.3)$$

Відмітимо, що якщо x_i має вільний член, з'явиться проблема ідентифікованості – неможливо обчислити λ і β_0 . З ряду причин (зручність, простота, чисельна стійкість, точність наближених процедур виведення), переважно краще оцінювати β_0 ніж λ , тому будемо використовувати цю параметризацію. Звичайно, отримавши оцінку β_0 , можна легко оцінити значення та довірчі інтервали для λ через перетворення $\lambda = \exp(\beta_0)$.

Оцінка β за методом максимальної правдоподібності є складною у випадку експоненційної регресії через те, що потрібно розв'язувати нелінійну систему рівнянь. Воно не може бути розв'язане у явній формі, для розв'язання потрібна ітеративна процедура.

Основною ідеєю є створити лінійне наближення до нелінійної системи рівнянь, обчислити з неї $\hat{\beta}$, обчислити нове наближення і повторювати до збіжності.

Як і з іншими регресійними моделями, інтерпретація коефіцієнтів регресії стосується ефекту від зміни одного з факторів за умови, що інші лишаються незмінними.

Розглянемо гіпотетичне порівняння між двома індивідами, чії пре диктори є однаковими окрім змінної j , де вони відрізняються на $\delta_j = x_{1j} - x_{2j}$. Тоді:

Для будь-якої моделі пропорційних ризиків, $\lambda_1(t)/\lambda_2(t)$ є константою, незмінною з часом. Ця константа (2.18) відома як відношення ризиків, або відносний ризик. Таким чином, інтерпретацією коефіцієнта регресії в моделі пропорційних ризиків є те, що $e^{\delta\beta}$ є відношення ризиків для зміни в δ одиниць в коваріаті.

Позначимо лінійний предиктор $\eta_i = x_i^T \beta$.

Враховуючи незалежне цензурування і вважаючи, що реальний час життя $\tilde{T}_i | x_i \sim \text{Exp}(\lambda_i)$, внесок логарифма правдоподібності i -го суб'єкта в експоненційну регресію наступний [3]:

$$l_i(\eta_i) = d_i \eta_i - t_i e^{\eta_i}, \quad (2.3)$$

де $d_i = 1\{\tilde{t}_i \leq c_i\}$, $t_i = \tilde{t}_i \wedge c_i = \min(\tilde{t}_i, c_i)$, c_i – час цензурування i -го суб'єкта, \tilde{t}_i – реальний час життя i -го суб'єкта.

Тоді похідна логарифму правдоподібності і гессіан мають наступний вигляд:

$$u_i(\eta_i) = d_i - t_i e^{\eta_i} \quad (2.4)$$

$$H_i(\eta_i) = -t_i e^{\eta_i} \quad (2.5)$$

Нехай μ позначає вектор, у якого i -й елемент дорівнює $t_i e^{\eta_i}$ і W позначає діагональну матрицю з i -м діагональним елементом $t_i e^{\eta_i}$. Тоді можна переписати похідну логарифму правдоподібності і гессіан у вигляді:

$$u(\eta) = d - \mu \quad (2.6)$$

$$H(\eta) = -W \quad (2.7)$$

Як було відзначено раніше, вирішення для $\mu = 0$ є складним, оскільки μ є нелінійною функцією від η . Тому, розглянемо наближення у вигляді ряду Тейлора

$$u(\eta) \approx u(\tilde{\eta}) + H(\tilde{\eta})(\eta - \tilde{\eta}) = d - \mu + W(\tilde{\eta} - \eta) \quad (2.8)$$

де μ і W фіксовані на $\tilde{\eta}$.

Підставляючи $\eta = X\beta$ у попереднє рівняння і розв'язуючи для β , отримуємо

$$\hat{\beta} \leftarrow (X^T W X)^{-1} X^T (d - \mu) + \tilde{\beta} \quad (2.9)$$

Так як маємо ітеративну процедуру, ми не отримуємо точне значення $\hat{\beta}$. Натомість, потрібно обчислити значення $\hat{\beta}$, перерахувати μ і W і повторювати до збіжності.

Алгоритм Ньютона-Рафсона збігається до оцінки максимальної правдоподібності (хоча це не гарантовано), так як правдоподібність логарифмічно увігнута і зростаюча, і це також вірно (зазвичай) для експоненційної регресії.

Алгоритм у вигляді блок-схеми зображений на рисунку 2.1.

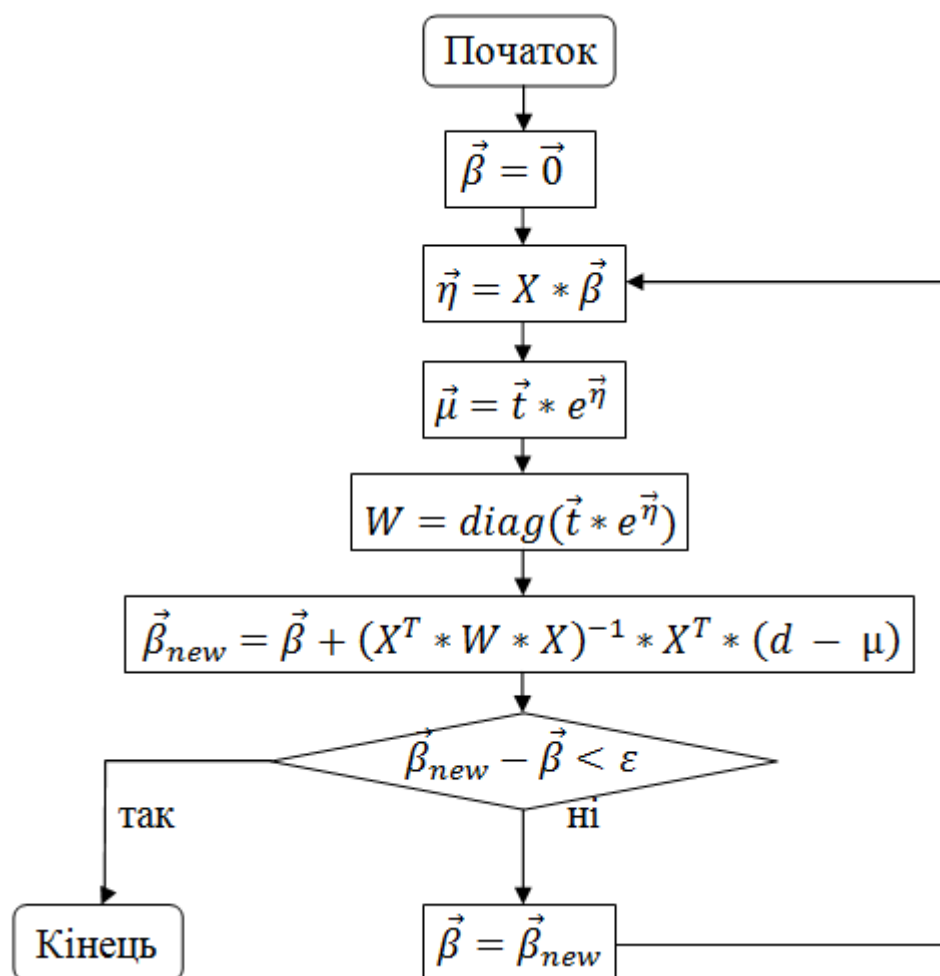


Рисунок 2.1 - Блок-схема алгоритму Ньютона-Рафсона

Існує три підходи до отримання довірчих інтервалів параметрів моделей виживання: заснований на похідній логарифму правдоподібності, метод Вальда і метод відношення правдоподібностей [3].

У методі, заснованому на похідній логарифму правдоподібності, перевіряється гіпотеза $H_0 : \theta = \theta_0$. Для цього обчислюється статистика

$$\frac{U(\theta_0)}{\sqrt{I(\theta_0)}} \quad (2.10)$$

і її значення порівнюється з нормальним розподілом. Як і звичайно, перетворюючи цей тест до $\alpha = 0.05$, можна отримати 95% довірчі інтервали для θ . Відмітимо, що для цього підходу, на відміну від двох наступних, не потрібно оцінювати значення θ .

Метод Вальда заснований на наближенні до похідної логарифму правдоподібності ряду Тейлора оцінки максимальної правдоподібності (H – матриця Гессе):

$$u(\theta) \approx H(\hat{\theta})(\theta - \hat{\theta}) \quad (2.11)$$

Таким чином,

$$\mathcal{J}^{\frac{1}{2}}(\hat{\theta} - \theta_0) \sim N(0,1), \text{ або } \hat{\theta} \sim N(\theta_0, \mathcal{J}^{-1}) \quad (2.12)$$

Тоді оцінка максимальної правдоподібності наближено нормальна, має середнє значення рівне справжньому значенню параметра і варіацію рівну оберненому значенню від інформації.

На основі цього результату, можна легко отримати тести або довірчі інтервали для θ .

Також розглянемо асимптотичний розподіл відношення правдоподібностей. Цей підхід також застосовує розклад в ряд Тейлора, але тут наближується логарифм правдоподібності.

$$l(\theta) \approx l(\hat{\theta}) + \frac{1}{2}(\theta - \hat{\theta})^T H(\hat{\theta})(\theta - \hat{\theta}) \quad (2.13)$$

Таким чином,

$$2\{l(\hat{\theta}) - l(\theta_0)\} \sim \chi_p^2 \quad (2.14)$$

Отримана за алгоритмом Ньютона-Рафсона оцінка $\hat{\beta}$ є оцінкою максимальної правдоподібності, за методом Вальда $\hat{\beta} \sim N(\beta, I^{-1})$. Залишилось лише отримати матрицю інформації відносно β . Продовжуючи попередні міркування, маємо:

$$\hat{\beta} \sim N(\beta, (X^T W X)^{-1}) \quad (2.15)$$

Звідси отримуємо довірчі інтервали для β_j :

$$\beta_j \pm z_{1-\frac{\alpha}{2}} SE_j \quad (2.16)$$

$$SE_j = \sqrt{(X^T W X)^{-1}_{jj}} \quad (2.17)$$

Підхід відношення правдоподібностей на практиці дещо ускладнюється у випадку декількох параметрів, оскільки ми не маємо оцінок параметрів у явних формах. Якби деякий параметр β_j був єдиним параметром, проблема би зводилась до знаходження коренів, де ми б обчислили довірчий інтервал за правилом $2(l(\hat{\beta}_j) - l(\beta_j)) = \chi_{1,95}^2$.

Проте, β_j не є єдиним параметром, тому всі оцінки максимальної правдоподібності відповідно зміняться. Іншими словами, оцінювання $l(\beta_j)$ не є простим, тому що потрібно перераховувати $\hat{\beta}_{-j}$ для кожного значення β_j для якого ми намагаємось знайти корені.

Правдоподібність $L(\beta_j, \hat{\beta}_j(\beta_j))$ називається профілюючою правдоподібністю, а процедура перерахування називається профілюванням.

Отримання довірчих інтервалів за допомогою похідної логарифму правдоподібності або відношення правдоподібностей вимагають профілювання, на відміну від підходу Вальда. На практиці, швидше і зручніше використовувати довірчі інтервали Вальда [3]. У бібліотеці `survival` на мові програмування R застосовуються саме вони.

Для випадкової величини часу до події T , модель прискорення часу відмови [3] пропонує наступне відношення між коваріатами та $Y = \log T$:

$$Y_i = x_i^T \beta + W_i, \quad (2.19)$$

де $W_i \sim^{iid} f$ є похибкою (або залишком). Такі моделі часто називаються лог-лінійними моделями. Вищенаведені позначення описують широкий клас моделей: в залежності від розподілу, який буде вказано для W , будуть отримані різні моделі, але всі вони будуть мати схожу загальну структуру.

Очевидно, можна припустити, що $W_i \sim^{iid} N(0, \sigma^2)$. Припущення про те, що Y має нормальний розподіл, еквівалентне припущенню, що T має логнормальний розподіл. Тоді, за відсутності цензурування, можна просто використати звичайний метод найменших квадратів, щоб побудувати модель, отримати довірчі інтервали, тощо. Але звичайно, майже завжди у дослідженнях присутнє цензурування, тому потрібно розширити звичайні лінійні методи так, щоб вони справлялись з цензуруванням. Більше того, логнормальний розподіл, незважаючи на свою зручність, не описує точно більшість розподілів часу до події.

Для будь-якої моделі прискорення часу відмови, маємо:

$$T = e^{\eta_i} T_0, \quad (2.20)$$

де $T_0 = e^W$ і $\eta_i = x_i^T \beta$.

Іншими словами, в той час як в моделі пропорційних ризиків коваріати впливають мультиплікативно на ризик, в моделі прискорення часу відмови коваріати впливають мультиплікативно на час до виникнення події.

$$\text{Функція виживання: } S_i(t) = S_0(e^{-\eta_i t}).$$

$$\text{Функція ризику: } \lambda_i(t) = \lambda_0(e^{-\eta_i t})e^{-\eta_i}.$$

Якщо порівнювати модель пропорційних ризиків і модель прискорення часу відмови на графіку логарифм часу проти логарифму ризику, ефект припущення пропорційних ризиків впливає на зміну ризику і спричиняє вертикальне зміщення, в той час як ефект моделі прискорення часу відмови спричиняє горизонтальний зсув. В загальному, дві моделі не можуть бути приведені одна до одної, тому процес може бути краще описаний лише однією з моделей, а не обома одночасно [3].

Проте, існує виключення: якщо розподіл є лінійним (на шкалі логарифму часу проти логарифму ризику), тоді будь-якому горизонтальному зміщенню лінії можна поставити у відповідність вертикальне. У цьому масштабі є лінійним розподіл екстремальних значень, а також сімейство розподілів Вейбулла. Тому, розподіл Вейбулла є єдиним розподілом, який задовольняє обидві моделі: пропорційних ризиків і прискорення часу відмови [3].

Згідно з вищенаведеним, вплив зміни на δ_j одиниць в коваріаті j призводить до збільшення часу відмови у $\exp(\delta_i \beta_i)$ разів.

2.2 Сегментація ліній тексту

Одним з найпоширеніших підходів до сегментації видобутку рядків у друкованих документах є метод проектування. Він також був адаптований для рукописних документів з невеликим перекриттям [1]. Цей простий спосіб підраховує кількість чорних пікселів у кожному рядку зображення. Ті лінії, що містять кількість пікселів нижче певного значення, вважаються лініями без

тексту, а інші - як текстові. Відхиленням від технології проєкційного профілю є згладжений проєкційний профіль [8], який використовує фільтр низьких частот для зменшення шуму. Інший цікавий метод, який застосовується до багатокожних рукописних документів для вилучення текстових ліній компонентів. Цей алгоритм передбачає, що гіпотетична вода тече з лівого та правого боків зображення, а символи текстових ліній працюють як дамба, яка блокує воду. Отримані ділянки, що залишилися незміщені на зображенні, нарешті позначаються як кандидати на текстові рядки регіонів. Однак цей спосіб потребує використання структурованого елемента, який визначає кут нахилу для затоплення, який залежить від лінійних схилів кожного конкретного документа. Наприклад застосування цього методу до зразка рукописного документа. Оцінка відповідного значення параметра (кут нахилу необхідного структурованого елемента) для кожного документа є складною і визначає обчислювальні показники роботи методу. Зразок двоетапного документа та (справа) сегментація лінії гіпотетичним потоком води. Сіримі областями є затоплені частини документа. Створення бінарної карти переходу зображення зображення для пошуку імовірного розташування текстових ліній застосовується в [2]. Пізніше виявлені лінії очищаються за допомогою алгоритму розрізання графіків з мінімальним розрахунком / максимальним потоком. Інший підхід використовує перетворення Nough на основі блоків для виявлення та вилучення необмеженої лінії [10]. Ніколас та інші запропонували метод сегментації ліній у документах, який базується на структурі вирішення проблем ІА виробничих систем. Лі та інші оцінюють карту вірогідності зображення документа, де кожен компонент представляє ймовірність того, що розглянутий піксель належить до текстової лінії. Вони розвивають початковий виявлення тексту та використовують метод встановлення рівня, щоб визначити межу сусідніх текстових ліній. Їхні експерименти показали, що запропонований алгоритм є надійним для масштабування зміни, обертання та шуму. У недавньому документі використовується запропонований метод вилучення текстових рядків на основі морфології для виявлення текстових областей у

зображеннях. Для обробки з перекосом рукописних текстових рядків автори застосовують метод на основі моментів для оцінки лінійних орієнтацій. Лікформан-Сулем та інші опублікував у 2007 році недавній оглядовий документ про текстову лінію (друковане та рукописне), сегментація історичних зображень документа. Ця робота детально описує складність цієї проблеми, а також описує класифікацію методів сегментації.

Процес вилучення рядків із документа використовується як основа для вилучення структури документів, розпізнавання рукописного тексту або вдосконалення тексту. Існує безліч методів, які стосуються проблеми вилучення рядка друкованого документа, яка, як правило, скорочується до глобального пошуку косусів (текстові рядки паралельні один одному, але не обов'язково горизонтальні). З іншого боку, при роботі з рукописним документом проблема стає більш складною лінії не паралельні один одному, однакові літери не мають однакових розмірів, текстові лінії мають літери, які поширюються на інші текстові лінії, висока текстова організація не може бути визначена (абзаци, підрозділи тощо). У будь-якому зображенні, документі чи ні, з рукописним або друкованим текстом кожен піксель може бути пов'язаний з важливістю (тобто, скільки пікселя впливає на загальне зображення).

2.3 Опрацювання шуму на зображеннях

Зображення часто деградуються шумами. Шум може виникнути під час захоплення зображень, передачі тощо. Вилучення шумів є важливим завданням при обробці зображень. Загалом, результати шумозаглушення сильно впливають на якість обробки зображень. Кілька способів видалення шумів добре зарекомендували себе при обробці кольорових зображень. Характер проблеми видалення шумів залежить від типу шуму, що руйнує зображення. У сфері зменшення шумів зображення було запропоновано декілька методів

лінійного та нелінійного фільтрування. Лінійні фільтри не здатні ефективно усунути імпульсний шум, оскільки вони мають тенденцію до розмивання країв зображення. З іншого боку, нелінійні фільтри підходять для боротьби з імпульсним шумом. За останні кілька років виникли кілька нелінійних фільтрів на основі класичної та нечіткої техніки. Наприклад, більшість класичних фільтрів, які одночасно видаляють розмивання країв, а нечіткі фільтри мають можливість поєднувати збереження та згладжування краю. У порівнянні з іншими нелінійними методами нечіткі фільтри здатні представляти знання зрозумілим способом.

У комп'ютерному природному сценічному аналізі широко використовуються кольорові зображення завдяки чудовим характеристикам над чорно-білими зображеннями. Розглянемо, наприклад, кілька сусідніх об'єктів з різним кольором, але однаковою світністю. Ці об'єкти можуть бути інтерпретовані як об'єднані об'єкти на графічному рівні зображення через постійні інтенсивності, але можуть бути відбиті яскраво в кольоровому зображенні. Додана інформація про кольори висвітлює будь-яку можливу двозначність. Окрім переносної інформації про відображену енергію, кольорові зображення містять спектральну інформацію про об'єкт та надають важливі деталі для швидкого візуального пошуку, перевірки та точної класифікації об'єктів. Більше того, колір поверхні відносно інваріантний для (інтенсивності) освітленості, що відбувається через властивість кольорового сталість. Тому обробка кольорової інформації є відносно послідовною, коли умови освітлення навколишнього середовища змінюються, і забезпечує більш надійні та точні результати для сприйняття машини та аналізу природного місця.

Будь-яке зображення, отримане оптичним, електрооптичним чи електронним способом, може бути деградоване недосконалістю механізмів зондування. Потенційні деградації можуть мати місце у вигляді сенсора, шумів фотографічного зерна, розмивання (нефокусована камера, рух відносно об'єктної камери), випадкова атмосферна турбулентність тощо. Наявність шуму

в зображеннях являє собою незворотну втрату інформації. Кілька фільтрів були розроблені для видалення шумів зображень.

Шум є результатом помилок у процесі отримання зображень, що призводить до значень пікселів, які не відповідають дійсності інтенсивності справжньої сцени. Зменшення шуму - це процес видалення шуму від сигналу. Методи зменшення шуму є концептуально дуже схожі, незалежно від оброблюваного сигналу, однак априорне знання характеристик очікуваного сигналу може означати, що реалізація цих прийомів сильно відрізняється залежно від типу сигналу. Зображення, зроблене датчиком, піддається фільтруванню різними фільтрами згладжування та отриманими зображеннями. Усі пристрої запису, як аналогові, так і цифрові, мають риси, які роблять їх чутливими до шуму. Основна проблема обробки зображень полягає в зменшенні шуму від цифрового кольорового зображення.

Удосконалення якості зображення було проблемою у всій області обробки зображень. Зображення постраждали від різних типів шумів. Шум зображення є небажаним, оскільки він погіршує якість зображення. Застосування зменшення шуму в обробці зображень є перспективним напрямом досліджень. Нечіткі технології вже застосовуються в декількох областях обробки зображень і мають численні практичні застосування. Звуковий шум, як правило, небажаний, різниця в яскравості або інформація про кольори розглядається як шум. Шум зображення може породжуватись у зерні плівок або в електронних шумах у датчику та схемі вхідного пристрою або в неминучому короткому шумі ідеального фотонного детектора. Шум зображення найбільш яскраво виражено в області зображень з низьким рівнем сигналу, такими як область тіні або під виглядом зображень.

Алгоритми вилучення зображень часто приймають додатковий білий гаусовий шум (AWGN), який не залежить від фактичних значень RGB. Такі підходи не є повністю автоматичними і не можуть ефективно видалити колірний шум, вироблений сучасною цифровою камерою ПЗЗ. У цьому документі ми пропонуємо єдину схему для двох завдань: автоматичну оцінку та

видалення кольорового шуму з одного зображення за допомогою кусочно гладких моделей зображень. Ми вводимо функцію рівня шуму (NLF), яка є безперервною функцією, яка описує рівень шуму як функцію яскравості зображення. Потім ми оцінюємо верхню межу реальної функції шуму, встановивши нижню огиду до стандартних відхилень відмінностей зображення на одному сегменті. Для видалення кольору кольоровий шум значно зменшується шляхом проєціювання значень пікселів на лінію, яка відповідає значенням RGB у кожному сегменті. Потім побудовано гауссовське умовне довільне поле (GCRF) для отримання нижчого чистого зображення із шумного вводу. Величезні експерименти проводяться для тестування пропонованого алгоритму, який, як показують, перевершує сучасні алгоритми стирання.

Розпізнавання зображень вивчалось десятиліттями в галузі комп'ютерного зору, обробки зображень та статистичної обробки сигналів. Ця проблема не тільки забезпечує хорошу платформу для вивчення природних моделей зображень та алгоритмів розподілу сигналів, але також стає важливою частиною систем придбання цифрових зображень для підвищення якості зображень. Ці два напрямки є важливими і будуть розглянуті в цьому документі.

Велика частина існуючої роботи з видалення зображення передбачає додавання білого гаусового шуму (AWGN) і видаляє шум, незалежний від каналів RGB. Однак тип і рівень шуму, який створюють цифрові камери, невідомі, якщо серія та марка камери, а також параметри камери (ISO, витримка затвора, діафрагма та вмикання / вимикання) невідомі. Наприклад, метадані формату файлів зображень (EXIF), додані до кожного зображення, можуть бути втрачені при перетворенні формату зображення та переміщенні файлу зображення. Тим часом, статистика колірному шуму не залежить від каналів RGB через демоверсійний процес, вбудований в камери. Отже, поточні підходи, що використовуються для відміни, не є справді автоматичними і не можуть ефективно видаляти кольоровий шум. Цей факт перешкоджає практичному

застосуванню методів видалення шуму до розпізнавання цифрових зображень та збільшення їх застосування.

У деякому програмному забезпеченні, що заперечує зображення, користувач повинен визначити кількість рівних зображень регіонів для оцінки рівня шуму. Це спонукало нас прийняти сегментаційний підхід для автоматичного визначення рівня шуму з одного зображення. Оскільки рівень шуму залежить від яскравості зображення, ми пропонуємо оцінити верхню межу функції рівня шуму (NLF) з зображення. Зображення поділяється на кусочно-гладкі області, в яких середнє значення має оцінка яскравості, а стандартне відхилення - переоцінка рівня шуму. Попередні функції рівня шумів вивчаються шляхом імітації процесу цифрової обробки зображень, і вони використовуються для правильної оцінки кривої, коли відсутні дані.

Оскільки відокремлення сигналу та шуму від одного входу є недостатньо обмеженим, теоретично неможливо повністю відновити оригінальне зображення від шумового забрудненого спостереження. Метою вилучення зображень є якнайбільше збереження можливостей зображення при усуненні шуму. Існує ряд принципів, які ми хочемо зіставити при розробці алгоритмів розпізнавання зображень.

2.3 Згорткова нейронна мережа

Конволюційні нейронні мережі мають іншу архітектуру, ніж звичайні нейронні мережі. Регулярні нейронні мережі перетворюють вхід, виводячи його через серію прихованих шарів. Кожен шар складається з набору нейронів, де кожен шар повністю з'єднаний з усіма нейронами в шарі раніше. Нарешті, є останній повний з'єднаний шар - вихідний шар, який представляє прогнози.

Конволюційні нейронні мережі дещо різні. Перш за все, шари організовані в 3 вимірах: ширина, висота і глибина. Крім того, нейрони в одному шарі не з'єднуються з усіма нейронами в наступному шарі, але тільки в невеликій його

області. Нарешті, кінцевий результат буде зведено до єдиного вектору оцінок імовірності, організованого вздовж вимірювання глибини.

CNNs мають два компоненти:

- Частина "Приховані шари / вибір ознак";

У цій частині мережа виконуватиме ряд згорток і операцій об'єднання, під час яких виявляються функції. Якщо у вас є зображення зебри, це частина, в якій мережа визнає свої смуги, дві вуха і чотири ноги.

- Класифікаційна частина.

Тут цілком зв'язані шари слугуватимуть класифікатором поверх цих витягнутих функцій. Вони будуть призначати ймовірність того, що об'єкт на зображенні є тим, що передбачає його алгоритм.

CNN особливо корисні для класифікації та розпізнавання зображень. Вони мають дві основні частини: частину вилучення функції та класифікаційну частину.

Основною спеціальною технікою в CNN є згортка, де фільтр слайд над входом і об'єднує вхідне значення + значення фільтра на карті об'єктів. Врешті-решт, нашою метою є подача нових зображень на нашу CNN, щоб вона могла дати вірогідність об'єкту, який вважає, що він бачить, або описує зображення з текстом.

CNNs мають широке застосування в області розпізнавання зображень та відео, систем рекомендацій та обробки природної мови. У цій статті приклад, який я візьму, пов'язаний з Computer Vision. Проте основна концепція залишається незмінною і може бути застосована до будь-якого іншого випадку використання!

Для швидкого озвучення нейронних мереж, ось дуже чітко окреслена серія статей. CNNs, як і нейронні мережі, складаються з нейронів з навчальними вагами та упередженнями. Кожен нейрон отримує кілька входів, приймає зважену суму за них, передає його через функцію активації і реагує на вихід. Вся мережа має функцію втрат, і всі поради та хитрощі, які ми розробили для нейронних мереж, все ще застосовуються на CNN.

Глибока згорткова нейронна мережа призвела до результатів прориву в численних практичних завданнях з машинного навчання, таких як класифікація зображень у наборі даних ImageNet, керування політикою навчання в іграх PlayAtari або настільну гру Go і підписи до зображення. Багато хто з цих додатків спочатку виконують функцію `extraction`, а потім подають їх результати в `trainable classifier`. Малат, 2012 р. Ініціював математичний аналіз глибококонволюційних нейронних мереж для вилучення ознак.

2.4 Висновки до розділу

Були розглянуті сучасні методи обробки зображень, методи побудови математичних моделей машинного навчання, зокрема, обрано для реалізації метод згорткової нейронної мережі.

Після огляду існуючих кроків обробки і аналізу зображень було обрано така схема, яка складається з 8 кроків: бінаризації вихідного зображення, виявлення області тексту, сегментації текстових рядків, реконструкції символів та обробки шумів, моделі класифікатора символів, збагачення даних, сегментацію символів на основі класифікатора.

РОЗДІЛ 3 ЗАПРОПОНОВАНА СИСТЕМА ОБРОБКИ РУКОПИСНОГО ТЕКСТУ

3.1 Бінаризація вихідного зображення

Що стосується рукописних історичних документів, то етап попередньої обробки має важливе значення. Бінаризація є загальноприйнятою практикою для використання у випадку історичного набору даних [2]. Мета бінаризації полягає в тому, щоб отримати зображення, яке має лише дві можливі значення: чорний або білий для кожного пікселя.

У літературі методи бінаризації поділяються на глобальні або локальні (адаптивні) [3]. Глобальні методи використовують одичне порогове значення для відокремлення чорно-білих пікселів, тоді як декілька значень використовуються для визначення локального порогу на основі локальної області.

Обидва підходи були перевірені на двозначний пошук образів рукописів. Місцеві методи гаусівського адаптивного порогового та адаптивного середнього порогу показують незрозумілі результати, як показано на рисунках 3.1 і рис 3.2. Як було зазначено в літературі, ці методи, як локальні методи визначення порогу, є добрими у випадку різноманітних фонових компонентів, різного роду колір, текстура або яскравість зображення, видимість тексту [4]. Проте в нашому випадку це не вдається, тому що на зображенні присутні шуми, але яскравість зображення та якість текстури навпаки - досить хороші.

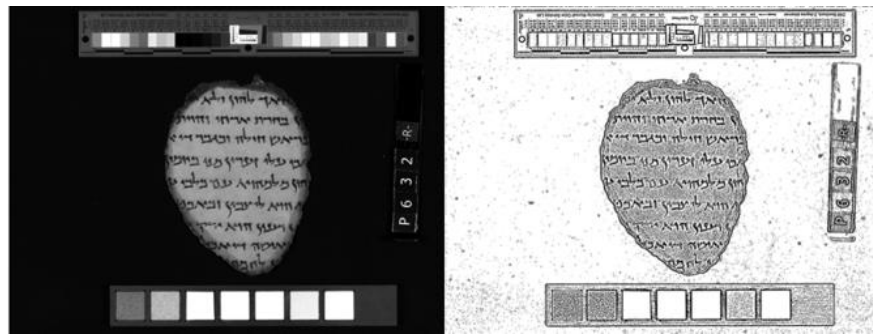


Рисунок 3.1 Адаптивна бінаризація середнього порогу. Вихідне зображення ліворуч, бінаризоване – праворуч

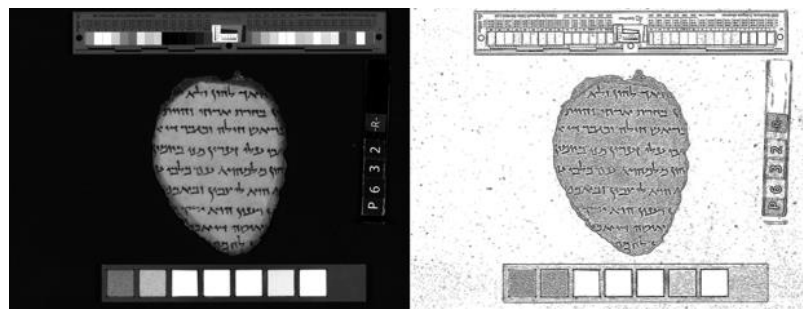


Рисунок 3.2 Гаусівська адаптивна бінаризація. Вихідне зображення ліворуч, бінаризоване – праворуч

Серед усіх методів бінаризації підхід OTSU [5] найкраще застосовується для шумних рукописів, оскільки він вибирає порогове значення, щоб мінімізувати дисперсію між класами порогових чорно-білих пікселів. Результат бінаризації OTSU показаний на рисинку 3.

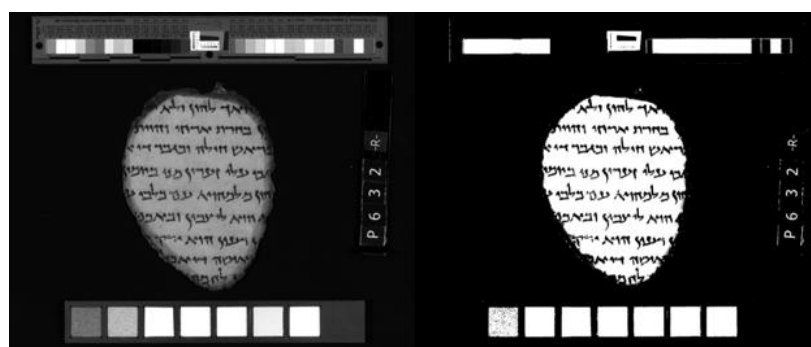


Рисунок 3.3 Бінаризація OTSU. Вихідне зображення ліворуч, бінаризоване – праворуч.

3.2 Знаходження та попередня обробка текстової області

Після бінаризації, наступний крок полягає в обрізанні області, де розташований текст. Оскільки зображення складаються з додаткової інформації, такої як код та кольорова палітра, цей крок знаходить частину зображення, де знаходиться текст. У цьому документі було проведено тестування двох методів для отримання текстової області:

1) Пов'язані компоненти;

Текстова область обрізана як найбільший з'єднаний компонент, як показано на рисунку 3.4. Метод має несправності у випадку перекриття символів або коли частина паперу не пов'язана з зображенням.

2) Контури.

Текстова ділянка обрізана як найбільший контур, як показано на рисунку 3.5. В основному цей метод знаходить криву, що з'єднує всі безперервні точки вздовж кордону. Як можна помітити, контурний метод дає кращі результати. Це вирішує проблеми, що з'явилися з попереднім методом.



Рисунок 3.4 Обрізаний текст за допомогою методу з'єднаних компонент

Причиною невдалого підходу підключених компонентів є відсутність незв'язаних частин рукопису, а також деякі збігаються букви.



Рисунок 3.5 Обрізаний текст за допомогою методу контурів

Врешті-решт, для основного алгоритму обрана методика контурів.

Під час цього кроку коректування фону здійснюється шляхом зміни кольору фону від чорного до білого. Фон вважається пікселями за межами знайденого контуру.

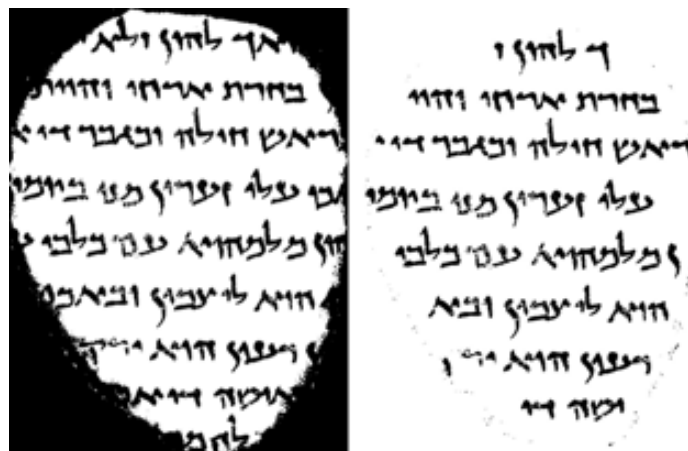


Рисунок 3.6 Обрізане зображення як найбільший контур - ліворуч, це ж зображення з налагодженим фоном справа

3.3 Сегментація ліній тексту

У літературі, часто трактується, що сегментація текстових рядків ґрунтується на проєкції гістограми [6], [7].

Виходом з попереднього кроку є зображення з чорними літерами та білим фоном. Прогнозування гістограми отримується підсумовуванням значень пікселів уздовж горизонтальної осі для кожного значення y .

На рисунку 7 ясно, що кожна текстова лінія відповідає піку в гістограмі. Пусті простори між піками представляють можливі області між різними текстовими рядками.

Проте іноді граф коливається, тому більше ніж один пік на рядок. Для вирішення цієї проблеми задаються порогові та мінімальні відстані. Таким чином, буде виявлено лише піки з амплітудою, більшою за поріг, і мінімальна відстань між кожним виявленим піком. Як тільки отримано піки, обчислюється середня висота лінії для пошуку координат сегментованих ліній.

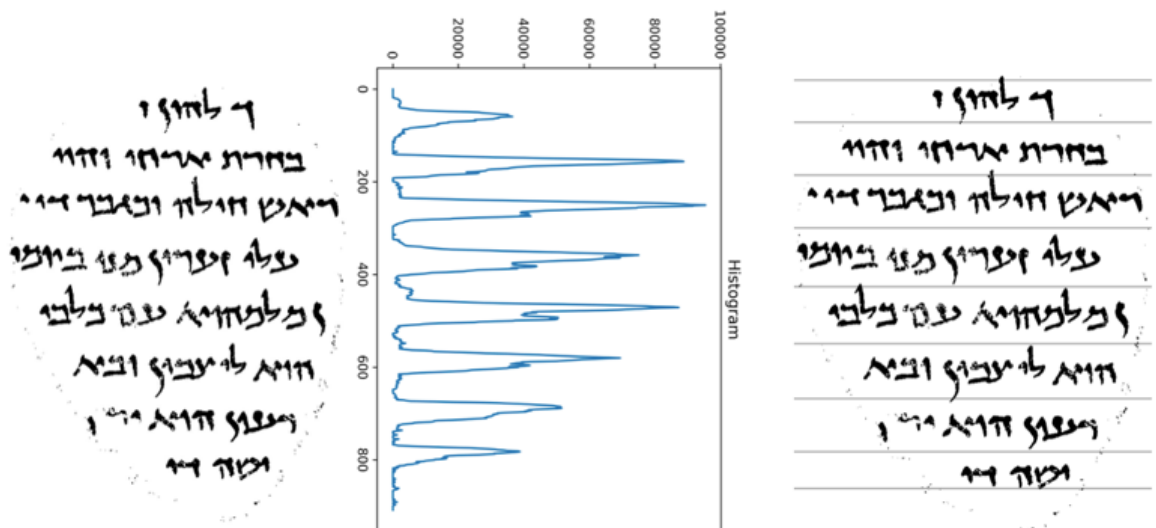


Рисунок 3.7 Сегментація ліній тексту за допомогою гісторамної проєкції

Слід зазначити, що запронована сегментація працює досить добре навіть із зображеннями низької якості.

3.4 Реконструкція символів та опрацювання шуму

Після того, як рядки сегментовані, зображення кожної лінії обробляється зверху на виявлених контурах, одночасно не включаючи контури менше встановленого порогу. При отриманні контурів символи на краях зображення спричинили проблему визначення всіх символів на краю як одного контуру. Це питання вирішується за допомогою простого та швидкого способу додати білі межі на 1 піксель на кожній стороні зображення. Як було зазначено вище, на цьому кроці наш спосіб обробки шумів полягає в тому, щоб ігнорувати їх під час виділення символів. Результат цього кроку показано на рисунку 8.

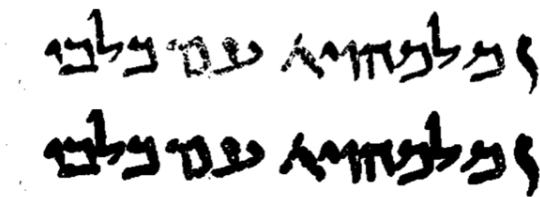


Рисунок 3.8 Реконструкція символів та опрацювання шуму, малюючи контур наверх

3.5 Модель-класифікатор літер

Модель CNN пройшла підготовку з набором івритських наборів рукописних цифр і перевірятиметься на зразок даних тестових даних того самого набору даних. Відсоток даних тесту з усього набору даних становить 30%. Кожна з вхідних зображень масштабується до розміру 32×32 .

Модель підготовлена на 10 епох. Розмір партії 256 для тренувань та тестування. У цій моделі використовується категоріальна кросентропія як функція втрат. Оптимізатор RMSprop використовується для оптимізації

навчального процесу. Серед 5116 зображень 3837 використовуються для навчання та 1279 використовуються для перевірки (25%).

Для реалізації CNN ми використовуємо 6 згорткових шарів і 1 повністю пов'язаний шар. Узор тричі аналогічний. Перший набір шарів може бути описаний наступним чином. Він має два згорткових шари з 32 фільтрами / ядрами з розміром вікна 3×3 , а потім максимальним об'ємом шарів з розміром вікна 2×2 . Також включено відсівний шар із коефіцієнтом відсіву 0,25, щоб уникнути переобладнання. У цьому випадку число ядра має такий самий порядок, що і ряд класів символів. Наступні два набори майже однакові, але вони мають 64 ядра. Нарешті, додається повністю з'єднаний щільний шар, а потім шар softmax, який виконує класифікацію серед 27 класів. Нижче наведено структуру моделі в таблиці 3.1.

Таблиця 3.1 – Структура згорткової нейронної мережі

Номер	Шар
1	2 Conv2D шарів з 32 ядрами з розміром вікна 3×3 , relu як функція активації
2	Max pooling шар з вікном 2×2
3	A dropout шар з dropout межою 0.25
4	2 Conv2D шари з 64 ядрами з вікном 3×3 , relu як функція активації
5	Max pooling шар з вікном 2×2
6	A dropout layer with a dropout ratio of 0.25
7	2 Conv2D layers with 64 kernels with a window size of 3×3 , relu as an activation function
8	A max pooling layer with window size 2×2
9	A dropout layer with a dropout ratio of 0.25
10	A dense layer which performs the classification among 27 classes with relu as an activation function
11	A dropout layer with a dropout ratio of 0.5
12	A dense layer with softmax as an activation function

Модель оцінювалася з використанням 10-кратної крос-валідації. Середнє значення втрати k-fold становить 0,3, а середнє значення точності k-fold становить 0,93. Криві втрати та точності наведені на рисунках 9 та 10.

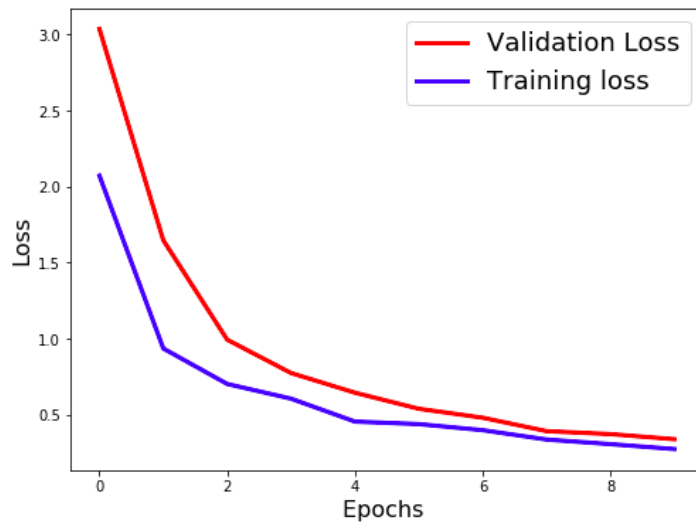


Рисунок 3.9 Крива втрат

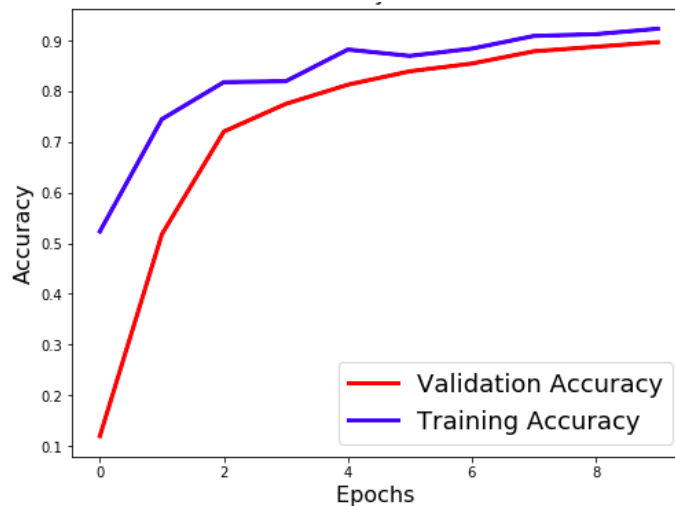


Рисунок 3.10 Крива точності

Оцінені втрати за даними тесту дорівнюють 0,5, тоді як точність дорівнює 0,85. Як інструмент реалізації моделі CNN використовувалася високорівнева нейронна мережа API Keras, написана на Python і здатна працювати на вершині TensorFlow з огляду на те, що вона розроблена з урахуванням можливості швидкого експерименту.

3.6 Збагачення даних

Збагачення даних перетворює нашу базову інформацію. У нашому випадку, він просто приймає наш набір даних зображень і перетворює їх шляхом обертання, масштабування, зсуву. Це робить нашу модель більш стійкою до переобладнання та чисельно збільшує розмір набору даних [8]. Крім того, він додає еластичні гауссові спотворення та випадкове стирання. Використання пружних спотворень імітує природний процес почерку [9].

Мета випадкового стирання полягає в тому, щоб робити моделі надійними для закупорки шляхом випадкового заміщення прямокутних областей з довільними значеннями пікселів. Процентна область зображення, що закривається випадковим прямокутником, становить 0,4 [10].

Обертання та зсув обмежуються певними кутами, які імітують перекося і рухомість, присутні в природному почерку. Максимальна кількість градусів для зсуву та обертання праворуч і ліворуч становить менше 10 градусів. Значення від 0 до 1, що представляє ймовірність виконання операції, встановлено на 0,5. Приклади збільшення даних зображені на рисунку 3.11.



Рисунок 3.11 Приклади збагачених даних

3.7 Розпізнавання літер за допомогою комплексного підходу (ковзного вікна і згорткової нейронної мережі)

Цілісний підхід використовується для сегментації символів та розпізнавання символів. Подібний підхід був запропонований Нильсоном [11] і Кейсі, Леколіне [12]. Ці два етапи об'єднані для проведення сегментації на основі визнання. Рішення щодо сегментації листа здійснюється на основі прогнозованої ймовірності за моделлю розпізнавання символів. Найвища ймовірність призводить до більш правильної сегментації. На цьому кроці образ попередньо сегментованої лінії подано у вигляді введення. Є два етапи, які повторюються для кожного потенційного персонажа. Перша - це генерація гіпотез сегментації однієї літери (горизонт). Другий - це вибір найкращої гіпотези (етап перевірки) [12]. Точніше, на першому етапі розмір вікна лінійного зображення дорівнює висоті рядка, оскільки існує припущення, що персонаж не має тенденцію бути ширшим, ніж це число. Розмір підволобу - це половина висоти рядка, якщо ця довжина є шириною літери. Розмір кроку становить 10 пікселів. Напрямок справа наліво, коли горизонтальна координата змінюється. Для кожного зображення під вікном вікна клас передбачається з його ймовірністю. На другому етапі з отриманого списку зображень під вікна приймається той, який має найвищу передбачувану ймовірність для цього вікна. Горизонтальна координата витягується з метою скорочення зображення та отримання залишку границі зображення як початкової точки до ітерації алгоритму ковзного вікна. Кожен наступний крок починається з витягнутого на попередньому кроці координати, зміщеної постійним ухилом, щоб запобігти розрізанню частин, що перекриваються. Ухилення дорівнює третині ширини підволобу. Це повторюється, тоді як залишкове зображення більше, ніж розмір підволобу. Розміри обох вікон і підсвічників постійні. Проте початкова координата підводячого проходу динамічно отримується на кожному етапі ітерації. Приклад

результатів розсувного вікна показано на рисунках 12 та 13. Можливості знайдених букв, як прогнозованих символів, складають 0,8 та 0,9, відповідно.

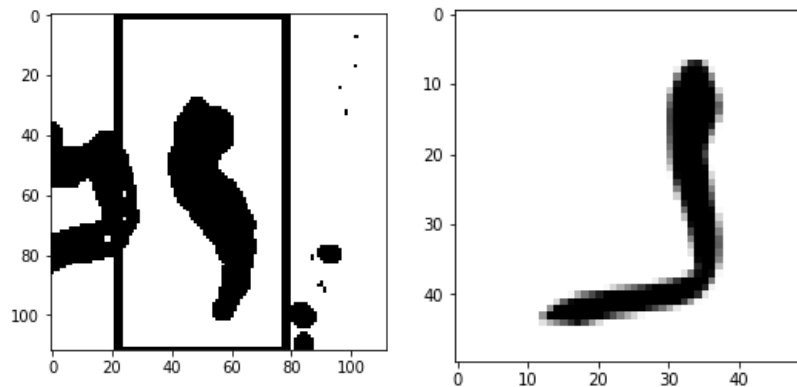


Рисунок 12 Результат кроку вікна з неперетинаючими літерами зліва і спрогнозованою літерою справа

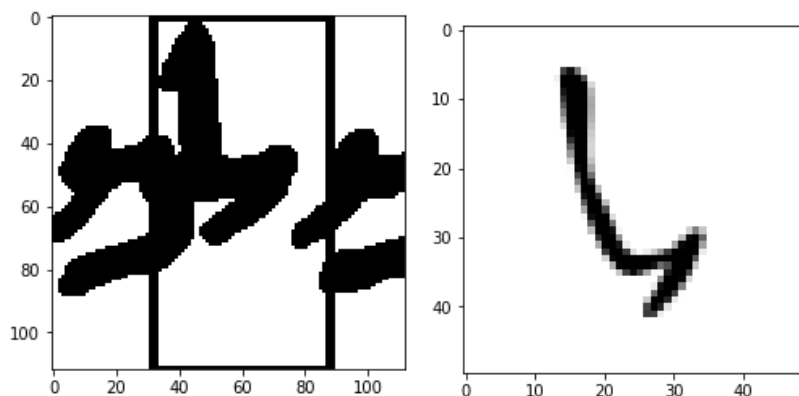


Рисунок 13 Результат кроку вікна з неперетинаючими літерами зліва і спрогнозованою літерою справа

З урахуванням виходу списку передбачуваних класів з його ймовірністю виконується етап фільтрування. Ймовірності менше 0,1 відфільтровані.

Перевага цього підходу полягає в тому, що символи прогножуються з відносно високою ймовірністю. Недоліком є те, що деякі символи пропускаються.

3.8 Оцінка моделі

Оцінка моделі подана у таблиці 3.1 за аналізом результатів відносно початкових даних і прогнозів за моделлю.

Таблиця 3.1 - Оцінка фактичних і передбачених значень

Фактичний клас	Передбачений клас	
	True(1)	False (0)
True (1)	TP – істинно позитивний	FN – хибно негативний
False (0)	FP – хибно позитивний	TN - істинно негативний

Для оцінки моделі використовують такі показники:

1. False positive rate – хибно позитивний показник

$$FPR = \frac{FP}{FP+TN} \cdot \quad (3.1)$$

2. Precision – акуратність

$$Precision = \frac{TP}{TP+FP} \cdot \quad (3.2)$$

3. Recall - відгук

$$Recall = \frac{TP}{TP+FN} \cdot \quad (3.3)$$

4. Accuracy – точність

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \cdot \quad (3.4)$$

3.9 Вибір програмної платформи

В останні роки машинне навчання перетворилося в тенденцію небувалої сили. Цей напрям підживлюється не лише дешевизною хмарних середовищ, але й доступністю найпотужніших процесорів та відеокарт, застосовуваних для подібних обчислень, — з'явилася ще і маса фреймворків для машинного навчання. Майже всі з них є відкритим програмним забезпеченням, але важливіше те, що ці фреймворки проектуються таким чином, щоб абстрагуватися від найважчих частин машинного навчання, роблячи ці технології більш доступними широкому класу розробників.

Машинне навчання — це наука змусити комп'ютер діяти, не будучи явно запрограмованим. В останнє десятиліття машинне навчання дало нам самокеровані автомобілі, розпізнавання мови, ефективний веб-пошук і значно покращене розуміння людського геному. Машинне навчання настільки розповсюджено в наш час, що ви можливо використовуєте його десятки разів на день, навіть не здогадуючись про це. Багато дослідників вважають, що це також найкращий шлях до штучного інтелекту, порівнянного з людським. У 1959 році Артур Самюель визначив машинне навчання як “Область науки, що надає комп'ютерам можливість навчатися, не будучи явно запрограмованими”. Є три основні причини необхідності машинного навчання, коли недостатньо просто запрограмувати комп'ютер. По-перше, розробники не можуть передбачити усі ситуації, в яких може виявитись машина. Наприклад робот, розроблений для проходження лабіринтів мусить дізнаватися план кожного нового лабіринту, який йому трапляється. По-друге, розробники не можуть передбачити усі зміни з часом: програма для прогнозування цін на фондовій біржі мусить навчатися для

адаптації, коли умови на ринку змінюються. По-третє, іноді люди не мають жодного уявлення про те, як запрограмувати розв'язок. Наприклад, більшість людей добре розпізнають обличчя своїх знайомих, але навіть найкращі програмісти не можуть створити програму для виконання цієї задачі, не використовуючи алгоритми навчання. Навчання, як і інтелект, охоплює такий широкий спектр процесів, що складно дати йому точне визначення. Словникові визначення включають такі фрази, як “отримувати знання, розуміння або вміння за допомогою вивчення, інструктування або досвіду” і “модифікація поведінкової тенденції через досвід.” Зоологи і психологи вивчають навчання людей і тварин. Існують паралелі між тваринним і машинним навчанням. Звичайно, багато технік у машинному навчанні походять від намагань психологів покращити свої теорії людського і тваринного навчання через обчислювальні моделі. Щодо машин, ми можемо сказати, дуже широко, що машина навчається коли змінює свою структуру, програму або дані (у відповідь на зовнішню інформацію) таким чином, що її очікувана майбутня продуктивність покращується. Машинне навчання зазвичай застосовується для змін у системах, що виконують завдання, пов'язані зі штучним інтелектом. Такі завдання охоплюють розпізнавання, діагностику, планування, контролювання роботів, прогнозування і таке інше. “Змінами” можуть бути або покращення вже робочих систем, або синтез цілковито нових [8].

Проведемо аналіз передових платформ для машинного навчання. Амазонівський сервіс машинного навчання надає інструменти візуалізації, які направляють користувачів через процес створення моделей машинного навчання без необхідності вивчати складні алгоритми і технології МН. Після того, як моделі будуть готові, цей веб-сервіс машинного навчання дозволяє легко отримати передбачення для застосування за допомогою простого прикладного програмного інтерфейсу. У Амазону є свій стандартний підхід до надання хмарних сервісів: спочатку зацікавленої аудиторії надається базова функціональність, ця аудиторія щось з неї ліпить, а компанія з'ясовує, що ж насправді потрібно людям.

Сервіс підключається до даних, що зберігаються в хмарі у базі даних чи у сховищі, він може виконувати двійкову класифікацію, багато класову категоризацію, а також регресію за вказаними даними для створення моделі. Мало того, що він використовує дані, що належать компанії сховищах, так ще і моделі не можна імпортувати або експортувати, а вибірки даних для тренувань не можуть бути більше 100 Гб. Але все ж це хороший інструмент для початку, ілюструє, що машинне навчання перетворюється з розкоші в практичний інструмент. Ілюстрація сервісу на рисунку 3.1.

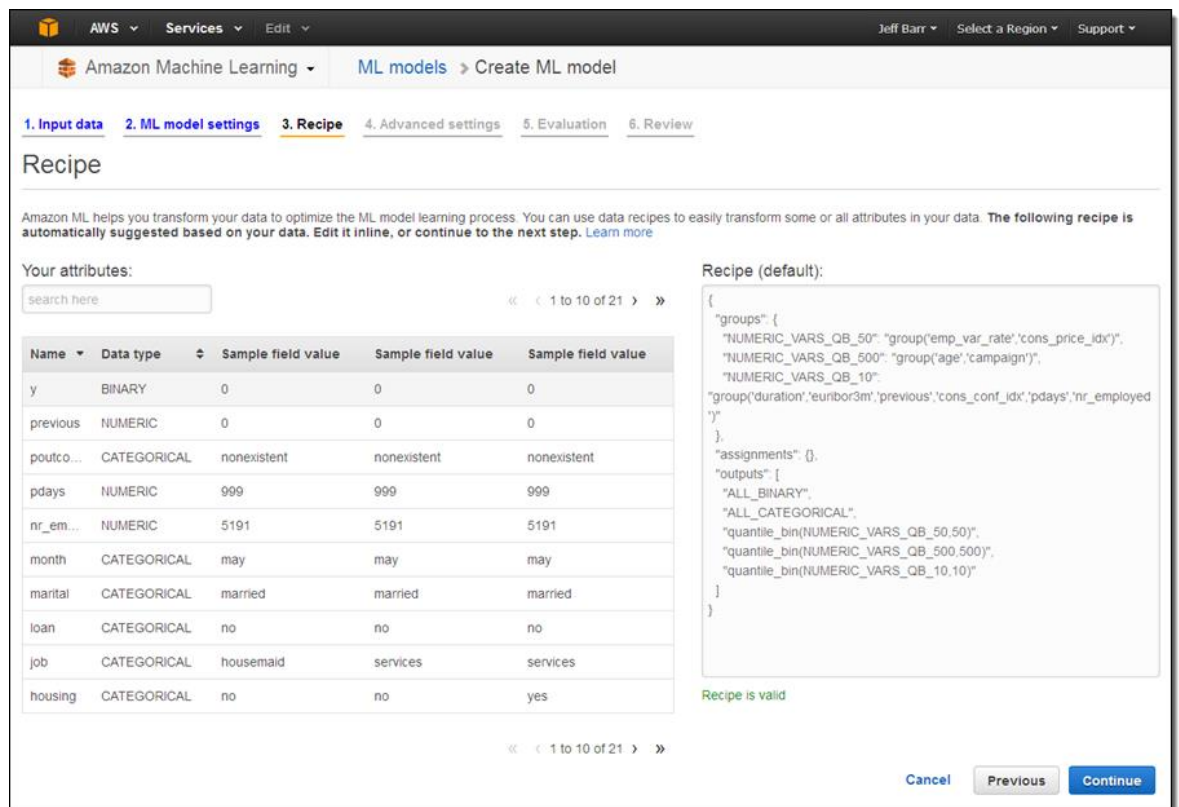


Рисунок 3.1 - Інтерфейс Амазонівського веб-сервісу машинного навчання

Майкрософтівська студія машинного навчання - Ажур. Враховуючи величезний обсяг даних і обчислювальної потужності, необхідний для машинного навчання, хмари є ідеальним середовищем для ML-додатків. Майкрософт оснастила Ажур власним сервісом машинного навчання, за який можна платити тільки за фактом використання — студія МН Ажур, рисунок 3.2. Доступні версії з помісячним і погодинною оплатою, а також безкоштовна. Зокрема, за допомогою цієї системи створено новий проект. Студія МН Ажур

дозволяє створювати і тренувати моделі, перетворювати їх з інтерфейсом для надання іншим сервісам. На один акаунт може виділятися до 10 Гб місця для зберігання даних, хоча можна підключити і власне Ажур-сховище. Доступний широкий спектр алгоритмів, створених Microsoft і сторонніми компаніями. Щоб спробувати сервіс, не треба навіть створювати акаунт, досить увійти анонімно, і можна використовувати студія МН Ажур протягом восьми годин.

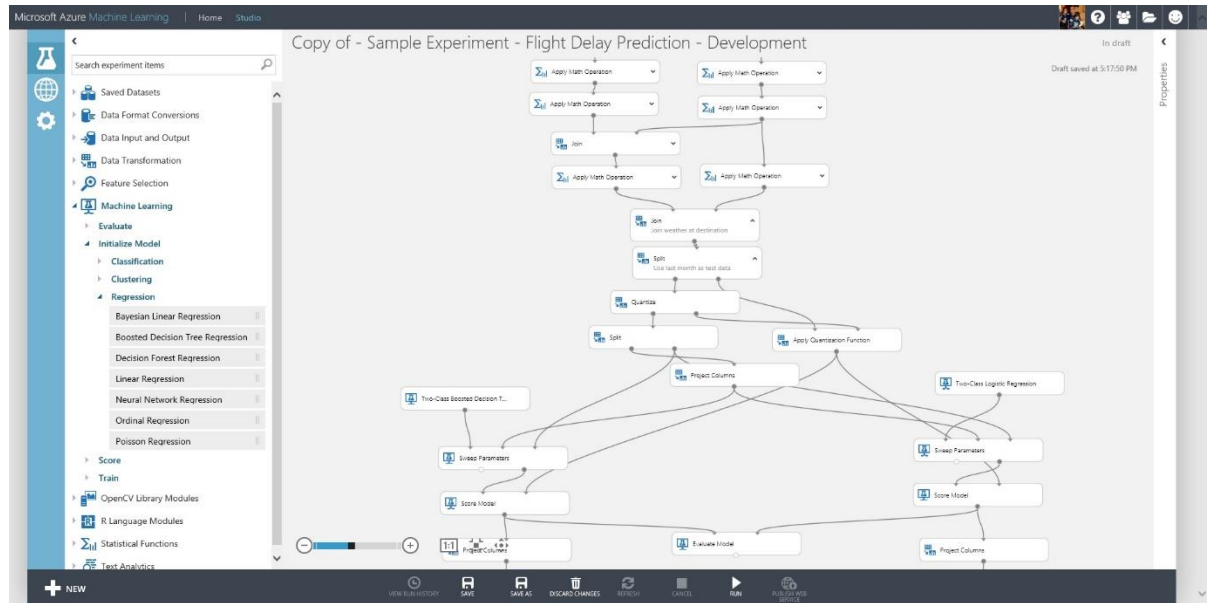


Рисунок 3.2 - Інтерфейс Майкрософтівською студії машинного навчання

Бібліотека МН Апаше Спарк. Апаше Спарк найбільше відомий завдяки своїй причетності до сімейства Хадуп. Але цей фреймворк для обробки даних всередині пам'яті з'явився поза Хадуп, і досі продовжує заробляти собі репутацію за межами цієї екосистеми. Спарк перетворився на звичний інструмент для машинного навчання завдяки зростаючій бібліотеці алгоритмів, які можна швидко застосувати до знаходяться в пам'яті.

Гуглівський тензор-флоу. Гуглівський тензор-флоу — це фреймворк машинного навчання, створений для розподілу обчислень в рамках кластера. Цей фреймворк розроблявся для вирішення внутрішніх проблем Гугл, але в кінці

кінців компанія випустила його у відкрите плавання у вигляді продукту з відкритим кодом.

Тензор-флоу реалізує графи потоків даних, коли порції даних («тензори») можуть оброблятися серією описаних графом алгоритмів. Переміщення даних по системі називається «потокими». Графи можна збиратися з допомогою С++ або Пайтон, і обробляти процесором або відеокартою. У Гугл є довгострокові плани розвитку Тензор-флоу силами сторонніх розробників, зображено на рисунку 3.3.

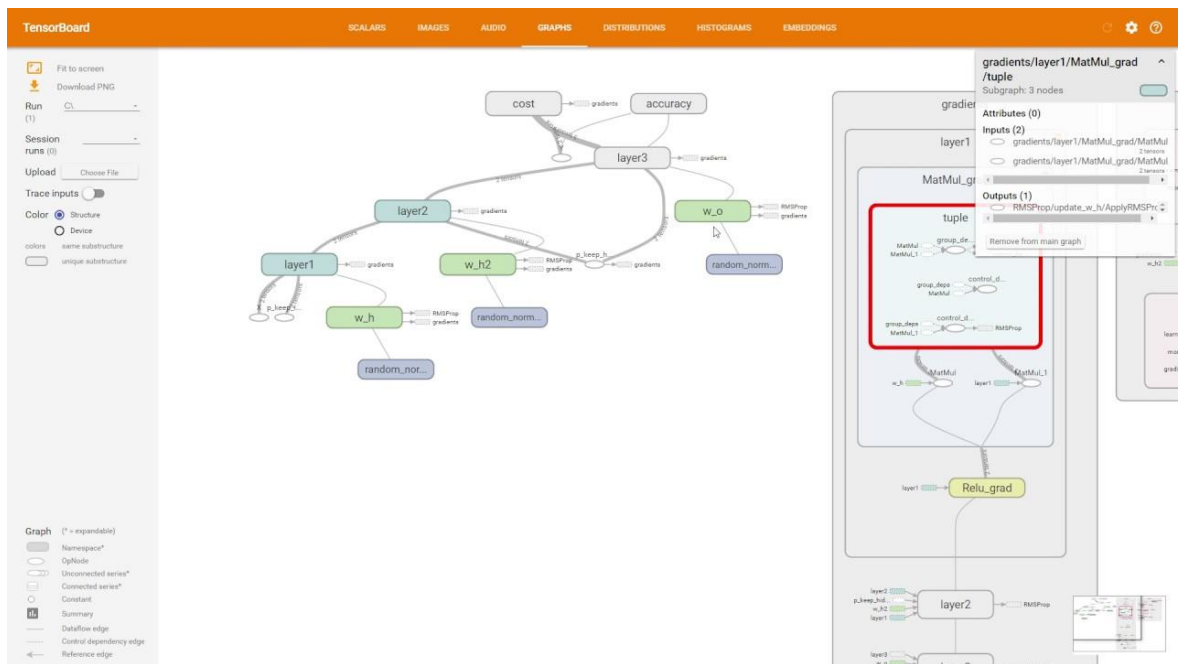


Рисунок 3.3 - Інтерфейс Гуглівського фреймворка машинного навчання

Велес — це розподілена платформа для створення програми глибокого навчання. Вона написана на С++, хоча для автоматизації та координації вузлів використовується Пайтон. Перш ніж згодуватися кластеру вибірки даних, їх можна аналізувати і автоматично нормалізувати. REST API дозволяє негайно використовувати натреновані моделі в робочих проектах (якщо у вас досить потужне обладнання).

Використання Пайтон у Велес виходить за межі «склеює коду». Наприклад, Пайтон (тепер Юпітер), інструмент для візуалізації та аналізу даних, може виводити дані з кластера Велес. Самсунг сподівається, що статус відкритого коду допоможе стимулювати подальший розвиток продукту, як і портування під Windows і Mac OS X.

3.10 Висновки до розділу

Був запропонований підхід системи розпізнавання рукописного тексту для історичних документів на мові іврит. Попередньої обробки включає в себе бінаризацію, пошук текстової області, подальшу обробку, такі як коригування фону, сегментацію ліній, реконструкцію символів та обробку шумів. Розпізнавання літер реалізований за допомогою згорткової нейронної мережі і підходу ковзного вікна. Реалізований програмний продукт з використанням мови програмування Python.

РОЗДІЛ 4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

4.1 Опис ідеї проекту

В межах підпункту послідовно проаналізовано та подано у вигляді таблиць:

- зміст ідеї;
- можливі напрямки застосування;
- основні вигоди, що може отримати користувач товару (за кожним напрямком застосування);
- чим відрізняється від існуючих аналогів та замінників;

Перші три пункти подано у вигляді таблиці (табл. 4.1) і дають цілісне уявлення про зміст ідеї та можливі базові потенційні ринки, в межах яких потрібно шукати групи потенційних клієнтів. [1]

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Додаток, що надає можливість прогнозувати динаміку фінансових ринків світу, в тому числі і зміну котирувань фондових індексів.	1. Використання в якості допоміжного сервісу, який допомагає прийняти вірне рішення.	Програмний додаток є набагато зручнішим для користувача (особливо якщо він не володіє необхідними знаннями).
	2. Прогнозування основних фінансових показників.	Кращі результати прогнозування динаміки

Аналіз потенційних техніко-економічних переваг ідеї (чим відрізняється від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів передбачає:

- визначення переліку техніко-економічних властивостей та характеристик ідеї;

- визначення попереднього кола конкурентів (проектів-конкурентів) або товарів-замінників, що вже існують на ринку, та проведення збір інформації щодо значень техніко- економічних показників.

Визначений перелік слабких, сильних та нейтральних характеристик та властивостей ідеї потенційного товару є підґрунтям для формування його конкурентоспроможності.

Таблиця 4.2. –Визначення сильних, слабких характеристик ідеї проекту

SEP

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	S (сильна сторона)
		Мій проект	Forecast 4Pr o	Fore xLin e	Evie ws		
1.	Можливість здійснювати коротко та довгострокові прогнози	+	-	+	-	Іноді прогноз може бути недостатньо точним	Мій проект дозволяє використовувати і вибирати кращі моделі, дає більше можливостей для керування
2.	Можливість подальшої роботи з отриманими даними не змінюючи середовища	+	-	-	+	Поки що варіація робіт досить незначна	На відміну від інших програм мій проект дозволяє обробляти отримані дані в зручному вигляді і взаємодіяти з ними

4.2 Технологічний аудит ідеї проекту

В межах даного підрозділу проведено аудит технології, за допомогою якої реалізувано ідею проекту.

Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових (табл. 4.3):

- за якою технологією буде виготовлено товар згідно ідеї проекту?
- чи існують такі технології, чи їх потрібно розробити/добробити?
- чи доступні такі технології авторам проекту?

Таблиця 4.3 – Технологічна здійсненність ідеї проекту [11]

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Оплата програми	Будь-яка платіжна система	Наявна	Доступна, у вигляді платних або безкоштовних бібліотек
2	Власне прогнозування потрібного процесу чи параметру	Навчання на статистичній вибірці моделей та їх порівняння	Частково: потрібно розробити частину, яка буде порівнювати моделі і обирати кращу або поєднувати декілька кращих	Бібліотеки існують у відкритому доступі, а бібліотека для вибору моделі ще має бути написана.
<p>Обрана технологія реалізації ідеї проекту: Так, проект можна починати реалізовувати. Спочатку варто реалізувати відсутню наразі бібліотеку з п. 2, а потім переходити до розробки власне додатку, адже саме вибір моделі є дуже важливим і основним для прогнозування.</p>				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку було проведено аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 4.4).

Середню норму рентабельності в галузі було порівняно із банківським відсотком на вкладення. Останній є меншим, тому є сенс вкладати гроші саме у цей проект.

Таблиця 4.4 – Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців (враховуючи вже існуючі і додатки), од	5-10
2	Загальний обсяг продаж, грн/ум.од	73551,59 млн грн.
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Необхідно вирішити питання з критерієм вибору моделі перш ніж виходити на ринок
5	Специфічні вимоги до стандартизації та сертифікації	-
6	Середня норма рентабельності в галузі (або по ринку), %	25-30%

За результатами аналізу таблиці 4.4 було зроблено висновок, що ринок є відносно привабливим для входження.

Надалі були визначені потенційні групи клієнтів, їх характеристики, та сформовано орієнтовний перелік вимог до товару для кожної групи (таблиця 4.5).

Після визначення потенційних груп клієнтів було проведено аналіз ринкового середовища: складено таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (таблиці 4.6, 4.7).

Таблиця 4.5. –Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Швидкий і зручний спосіб для оцінки та прогнозування динаміки фінансових інструментів.	Люди, які працюють з аналітикою ринку та прогнозуванням фінансових ринків, аналітики брокерських компаній, інвестиційні фонди, приватні інвестори.	Людина цінує свій час, прагне контролювати ситуацію, у людини є фінансові можливості для забезпечення своєї потреби в прогнозуванні.	Необхідно забезпечити доступність та простоту використання.
2	Більш ефективне використання моделей для оцінки та прогнозування шляхом комбінації результату.	Люди, яким важлива якість прогнозу та з попереднього пункту.	Мають інші програми для оцінки та прогнозування, хочуть мати гарантії щодо точності результату отриманого.	Необхідно розробити механізм гарантування цілісності та якості програми, а також систему оцінки моделей для інформування користувача

Після визначення потенційних груп клієнтів було проведено аналіз ринкового середовища: складено таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (таблиці 4.6, 4.7).

Ринкові можливості - це сприятливі обставини, які підприємство може використовувати для отримання переваг. Як приклад ринкових можливостей можна привести погіршення позицій конкурентів, різке зростання по-питу, появу нових технологій виробництва продукції, зростання рівня доходів населення і т. п. Слід зазначити, що можливостями з погляду SWOT-аналізу є не всі можливості, які існують на ринку, а тільки ті, які можна використовувати

Ринкові загрози - події, настання яких може несприятливо вплинути на підприємство. Приклади ринкових загроз: вихід на ринок нових конкурентів, зростання податків, зміна смаків покупців, зниження народжуваності й т. п.

Чинники попиту (тут доцільно взяти до уваги місткість ринку, темпи його зростання або скорочення, структуру попиту на продукцію підприємства і т. ін.);

Чинники конкуренції (слід врахувати кількість основних конкурентів, наявність на ринку товарів-замінників, висоту бар'єрів входу на ринок і виходу з нього, розподіл ринкових часток між основними учасниками ринку і т.ін.);

Чинники збуту (необхідно надати увагу кількості посередників, наявності сіток розподілу, умовам поставок матеріалів та комплектуючих і т. ін.);

Економічні чинники (враховується курс гривні (долара, євро), рівень інфляції, зміна рівня доходів населення, податкова політика держави і т.ін.); ^[L]_{SEP}

Політичні і правові чинники (оцінюється рівень політичної стабільності в країні, рівень правової письменності населення, рівень законслухняності, рівень корумпованості влади і т.ін.);

Науково-технічні чинники (звичайно береться до уваги рівень розвитку науки, ступінь упровадження інновацій (нових товарів, технологій) в промислове виробництво, рівень державної підтримки розвитку науки і т. ін.);

Соціально-демографічні чинники (слід врахувати чисельність і статеву та вікову структури населення регіону, в якому працює підприємство, рівень народжуваності і смертності, рівень зайнятості населення і т.ін.);

Таблиця 4.6 – Визначення ринкових можливостей і загроз

Параметри оцінки	Можливості	Загрози
1. Попит	Адаптація до ринку, що розширяється, проведення кампаній по залученню більшої кількості людей для користування додатком. Можлива реакція на загрозу: Переорієнтація продукту разом з переорієнтацією ринку, додавання можливості безкоштовного використання.	Темпи можливого скорочення ринку в деяких конкретних сегментах можуть свідчити про спадання інтересу до програм прогнозування та оцінки на користь вже існуючих перевірених методів
2. Конкуренція	Інноваційний підхід до прогнозування дозволить захопити досить велику частку ринку і потенційно розширити за рахунок людей, що надаватимуть перевагу якості Можлива реакція: Розгляд можливості покупки їх технології та доопрацювання якості виконання додатку, розробка нових властивостей, яких немає у конкурента.	Можливість появи конкурентів з дуже схожими функціями, які володіють необхідними і недоступними для нас зараз технологіями, і їх вихід на ринок раніше за нас
3. Науково-технічні	Можливий підйом науково-технічної спільноти до теми більш точного прогнозування рівня корупції, що дозволить швидше розповсюдити продукт.	Можлива не довіра наукової спільноти до використання методу. Для того, щоб мати аргументи і нароби ніхто не використав необхідно здійснити патентування
4. Соціально-культурні	Глобалізація суспільства приводить до того, що ми більше спостерігаємо за досвідом інших суспільств та переймаємо користі ініціативи	Недовіра науковим методам, можливе надання переваги більш перевіреним сподобам пригнічення корупції
5. Збут	На ринку з'явилася нова роздрібна сітка, яка в даний момент вибирає постачальників	З цього року наш найбільший оптовий покупець визначає постачальників за результатами тендеру

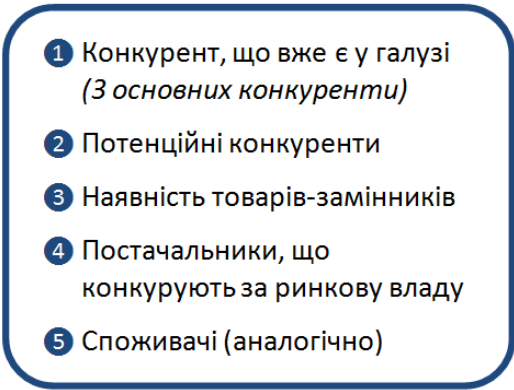
Надалі було проведено аналіз пропозиції: визначили загальні риси конкуренції на ринку (таблиця 4.7).

Таблиця 4.7 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - чиста конкуренція	Декілька основних гравців	Звертати увагу на якість послуг, що надаються, пропонувати послуги, які не пропонують конкуренти, оперативно обробляти канали зворотного зв'язку клієнтів з розробниками
2. За рівнем конкурентної боротьби - глобальне	Програма буде доступна в усіх країнах світу	Акцент на потреби держ діячів, аналітиків тощо
3. За галузевою ознакою - економічна	Конкуренцію складають програми для прогнозування	Акцентувати увагу на унікальності й універсальності послуг, що надаються
4. Конкуренція за видами послуг: - між бажаннями	Потенційні клієнти роблять вибір між бажанням швидше отримати бажаний результат та бажанням зробити це дешевше.	Чітко зрозуміти потреби та бажання кожної з груп цільової аудиторії, та розробляти гнучку систему, яка задовольнятиме потреби всіх груп користувачів
5. За характером конкурентних переваг - нецінова	Акцент знаходиться на незвичайності та якості послуг, що надаються, а також перевагам, які отримує клієнт при користуванні послугами.	Постійне покращення сервісу, залучення більшої бази знань, своєчасна обробка каналів зворотного зв'язку
6. За інтенсивністю - не марочна	Продається втілення ідеї, а не певний бренд	Просування ідеї програми у соціальних мережах, виховування спільноти, що свідомо та відповідально відноситься до аналізу та прогнозу.

Було проведено аналіз конкуренції у галузі за моделлю М. Портера (таблиця 4.8). М. Портер вирізняє п'ять основних факторів (Рисунок 4.1), що впливають на привабливість вибору ринку з огляду на характер конкуренції.

Рисунок 4.1 – Модель Портера

- 
- 1 Конкурент, що вже є у галузі
(3 основних конкуренти)
 - 2 Потенційні конкуренти
 - 3 Наявність товарів-замінників
 - 4 Постачальники, що конкурують за ринкову владу
 - 5 Споживачі (аналогічно)

Сильні позиції компанії за кожним з факторів означають її можливості забезпечити необхідні темпи обороту капіталу та її здатність впливати на інших агентів ринку, диктуючі їм власні умови співпраці. Характеристики факторів моделі відрізняються для різних галузей та змінюються із часом. Сила кожного фактору є функцією від структури галузі та її техніко-економічних характеристик.

Економічні чинники (враховується курс гривні (долара, євро), рівень інфляції, зміна рівня доходів населення, податкова політика держави і т.ін.);

Політичні і правові чинники (оцінюється рівень політичної стабільності в країні, рівень правової письменності населення, рівень законослухняності, рівень корумпованості влади і т.ін.);

Таблиця 4.8 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	Forecast4Pro, ForexLine, Eviews	Інші схожі розробки та програми (з такими ж функціями)	Рівень довіри користувачі в до нашого проекту	Кількість зацікавлених клієнтів, рівень зацікавленості в такому типі послуг	Поява схожих дешевших або якісніших продуктів-конкурентів
Висновки:	Є, але потенційно наш проект матиме більше можливостей і більший функціонал, що призначені саме для вирішення проблеми пригнічення корупції, тож питання тільки в якості і ціні.	Можливості входу в ринок присутні, необхідно вирішити проблему вибору моделі, потенційні конкуренти – можуть використати наші знання та помилки для створення кращого продукту	Успіх нашого проекту залежить від рівня довіри користувачі в до програми.	Клієнти формують попит на таку послугу.	Коли з'являться конкуренти чи товари-замінники, обробка каналів зворотнього зв'язку, розширення бази знань для прискорення роботи програми стане ключовим фактором успіху проекту.

За результатами аналізу таблиці 4.8 було зроблено висновок про можливість роботи на ринку з огляду на конкурентну ситуацію. Цей висновок був врахований при формулюванні переліку факторів конкурентноспроможності у наступному пункті.

На основі аналізу конкуренції, проведеного в таблиці 4.8, а також із урахуванням характеристик ідеї проекту (таблиця 4.2), вимог споживачів до товару (таблиця 4.5) та факторів маркетингового середовища (таблиці 4.6, 4.7)

визначається та обґрунтовується перелік факторів конкурентоспроможності. Аналіз оформлюється за таблицею 4.9.

Таблиця 4.9 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Фактор часу	Окремі частини концепції є новими, для перейняття ідеї та втілення її у життя потенційним конкурентам знадобиться час
2	Фактор новизни товару	Початковий успіх продукту очікується через його новизну та інтерес цільової аудиторії до нових інноваційних рішень

Таблиця 4.10 - Порівняльний аналіз сильних та слабких сторін «назва проекту»

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з Forecast_KD						
			-3	-2	-1	0	+1	+2	+3
1	Фактор часу	11	+						
2	Фактор новизни товару	15			+				

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (таблиця 4.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (таблиця 4.11).

Перелік ринкових загроз та ринкових можливостей було складено на основі аналізу факторів загроз та факторів можливостей маркетингового

середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення. Наприклад: зниження доходів потенційних споживачів – фактор загрози, на основі якого можна зробити прогноз щодо посилення значущості цінового фактору при виборі товару та відповідно, – цінової конкуренції (а це вже – ринкова загроза).

Таблиця 4.11 - SWOT- аналіз стартап-проекту

<p>Сильні сторони: Якість послуг, що надаються Новизна послуг Націленість на активних та успішних людей</p>	<p>Слабкі сторони: Поки що не вирішена проблема з критерієм вибору оптимальної моделі (точніше комбінації)</p>
<p>Можливості: Створення лідера серед сервісів прогнозування Прогрес у сфері гарантування результату (за рахунок високої точності) Потреба в розумному та ефективному використанні вже зібраних даних</p>	<p>Загрози: Падіння попиту на прогнозування динаміки фінансових ринків (малоймовірно) Посилення конкуренції на ринку</p>

На основі SWOT-аналізу було розроблено альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (див. таблицю 5.9, аналіз потенційних конкурентів). Визначені альтернативи були проаналізовані з точки зору строків та ймовірності отримання ресурсів (таблиця 4.12).

Таблиця 4.12 - Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
-------	--	--------------------------------	-------------------

1	<ul style="list-style-type: none"> - Ціль: отримання прибутку в короткостроковій перспективі - Конкуренція: цінова (пропонуємо свої нові послуги дешевше, ніж конкуренти) - Взаємодія з фірмами: активна боротьба за долю ринку, що належить конкурентам 	<p>В короткостроковому плані – велика</p> <p>В довгостроковому плані – значний ризик втратити долю ринку, якщо займатися лише ціновою конкуренцією. В основному треба зосередитись на якості.</p>	<p>Місяць</p> <p>- рік після запуску проекту.</p>
2	<ul style="list-style-type: none"> - Ціль: захоплення частини ринку, підтримання її розміру та поступове нарощення об'ємів - Конкуренція: нецінова (акцент на тому, що пропонуємо якісні і більш точні прогнози) - Взаємодія з конкурентами: співпраця, активний моніторинг їх діяльності, при можливій появі реальних конкурентів можна запропонувати злиття компаній/проектів 	<p>Висока імовірність отримання ресурсів та утримання їх протягом довгого проміжку часу. Більш імовірний розвиток компанії та постійне покращення продукту.</p>	<p>Місяць-рік після запуску проекту – для отримання перших фінансових надходжень від продажів.</p>

Обрано альтернативу 2 як таку, що має на увазі довше життя проекту. З означених альтернатив обирається та, для якої: а) отримання ресурсів є більш простим та ймовірним; б) строки реалізації – більш стислими.

Враховуючи, що впровадження онтології є головним завданням поліпшення спільної роботи смар-елементів розумного будинку, а час виконання однаковий, то обираємо перший варіант.

4.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: було проведено опис цільових груп потенційних споживачів (таблиця 4.13).

Таблиця 4.13 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Ділові люди до 50 років, що мають потребу в прогнозуванні фінансових ринків	Споживачі, як досить сучасні та знайомі з технологіями люди, готові сприйняти продукт	Потенційно високий, адже ділові люди зазвичай цінують свій час, а наш продукт дозволяє його економити	Є невелика конкуренція, але всі методи і моделі розділені в різних продуктах, тож наш проект матиме великий вплив	При наявності достойної та доречної реклами – досить просто

Продовження таблиці 4.13

№ п/п	Опис профілю цільової	Готовність споживачів	Орієнтовний попит в межах	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
-------	-----------------------	-----------------------	---------------------------	--------------------------------------	--------------------------

	групи потенційних клієнтів	сприйняти продукт	цільової групи (сегменту)		
2	Ділові люди старше 50 років, що мають потребу в прогнозуванні і фінансових ринків	Споживачі недостатньо готові сприйняти продукт, так як більшість з них, скоріш за все, вже мають готове рішення і не хочуть розбиратися в новому	Потенційно не надто високий	Невелика конкуренція, більшість з таких людей вже використовують якийсь сервіс.	Досить складно, адже потрібно переконати людей відмовитися від старих рішень та перейти на наш продукт
3	Нові люди в області інвестицій у фінансові ринки, що тільки шукають потрібний сервіс	Споживачі готові сприйняти продукт через його простоту та можливість, які він надає.	Високий попит	Присутня, але конкуренти неактивні.	Значна простота входу у сегмент через відсутність активної конкуренції та готовність споживачів до продукту.
Які цільові групи обрано: 3,1					

За результатами аналізу потенційних груп споживачів було обрано цільову групу, для якої буде запропоновано даний товар, та визначено стратегію охоплення ринку - стратегію концентрованого маркетингу (компанія зосереджується на одному сегменті).

Для роботи в обраних сегментах ринку сформовано базову стратегію розвитку (таблиця 4.15).

За М. Портером, існують три базові стратегії розвитку, що відрізняються за ступенем охоплення цільового ринку та типом конкурентної переваги, що має бути реалізована на ринку (за витратами або визначними якостями товару).

Стратегія лідерства по витратах передбачає, що компанія за рахунок чинників внутрішнього і/або зовнішнього середовища може забезпечити більшу, ніж у конкурентів маржу між собівартістю товару і середньо-ринковою ціною (або ж ціною головного конкурента). Зокрема, ця стратегія припускає, що за рахунок великих можливостей по об'ємах збуту товарів (портфеля укладених контрактів на постачання) і продуктивності підприємство може добитися менших витрат. Ця стратегія зазвичай тісно пов'язана з можливістю досягнення ефекту масштабу і досвіду.

Компанії, що вибирають цю стратегію, проводять ретельний контроль за постійними витратами, знижують виробничі, збутові і рекламні витрати, проводять інвестиції, спрямовані на зменшення витрат, ретельне опрацювання конструкції нових товарів.

Переваги стратегії за Ж.-Ж. Ламбеном:

- фірма здатна протистояти своїм прямим конкурентам навіть у разі цінової війни і в змозі отримувати прибуток при ціні, мінімально допустимій для конкурентів;
- сильні клієнти не можуть добитися зниження ціни нижче рівня, прийняттого для найбільш сильного конкурента;
- низькі витрати забезпечують захист проти сильних постачальників, оскільки дають фірмі велику гнучкість у разі підвищення вхідних витрат;
- низькі витрати створюють бар'єр входу для нових конкурентів і одночасно хороший захист проти товарів-замінників. В ході конкурентної боротьби з використанням цієї стратегії з ринку вимушені

будуть піти фірми, менш ефективні з точки зору величини і структури витрат, нездібні до проведення технологічних новацій, спрямованих на зниження витрат. Стратегія диференціації передбачає надання товару важливих з точки зору споживача відмінних властивостей, які роблять товар відмінним від товарів конкурентів. Така відмінність може базуватися на об'єктивних або суб'єктивних, відчутних і невідчутних властивостях товару(у ширшому розумінні – комплексі маркетингу), бути реальною або уявною. Інструментом реалізації стратегії диференціації є ринкове позиціонування.

Переваги стратегії за Ж.-Ж. Ламбенем:

- По відношенню до прямих конкурентів диференціація знижує ступінь замінності товару, посилює прихильність марці, зменшує чутливість до ціни і тим самим підвищує рентабельність;
- Прихильність клієнтів послабляє їх тиск на фірму і перешкоджає приходу на ринок нових конкурентів;
- Підвищена рентабельність збільшує стійкість до можливого зростання витрат в результаті дій сильного постачальника;
- Відмінні властивості товару і завойована прихильність клієнтів захищають фірму і від товарів-замінників. Реалізація цієї стратегії вимагає, як правило, більш високих витрат. Проте успішна диференціація дозволяє компанії домогтись більшої рентабельності за рахунок того, що ринок готовий прийняти більш високу ціну (цінову премію бренду). При веденні конкурентної боротьби з використанням цієї стратегії на ринку в першу чергу терплять фіаско фірми, що не здатні визначати потреби цільових ринків, оперативно реагувати на зміни в

ринковому попиту, проводити ефективну політику маркетингових комунікацій, не мають необхідних навичок в області брендингу.

Стратегія спеціалізації передбачає концентрацію на потребах одного цільового сегменту, без прагнення охопити увесь ринок. Мета тут полягає в задоволенні потреб вибраного цільового сегменту краще, ніж конкуренти. Така стратегія може спиратися як на диференціацію, так і на лідерство по витратах, або і на те, і на інше, але тільки у рамках цільового сегменту. Проте низька ринкова доля у разі невдалої реалізації стратегії може істотно підірвати конкурентоспроможність компанії.

Таблиця 4.14 - Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Захоплення, підтримання та захист частки ринку	Стратегія концентрованого маркетингу	Новизна послуг Доступність продукту Простота в користуванні продуктом	Стратегія диференціації

Таблиця 4.15 - Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Так	Нові споживачі, частково забиратиме споживачів конкурентів	Ні	Стратегія лідера

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до продукту (див. таблицю 4.5), а також в залежності від обраної базової стратегії розвитку (таблиця 4.15) та стратегії конкурентної поведінки (таблиця 4.16) розроблено стратегію позиціонування (таблиця 4.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Позиціонування — це маркетингове забезпечення товарів бажаного місця на ринку і у свідомості потенційних покупців (образ). Позиція компанії чи продукту показує чим він унікальний УТП (унікальну торговельну пропозицію), чим відрізняється від конкурентів (відстройка від конкурентів), чим корисний споживачу.

З точки зору маркетингу товар являє собою сукупність відчутних (розмір, колір, маса, швидкість і т. п.) і невлених (престижність, модність, сучасність і т. п.) властивостей.

Стратегія спеціалізації передбачає концентрацію на потребах одного цільового сегменту, без прагнення охопити увесь ринок. Мета тут полягає в задоволенні потреб вибраного цільового сегменту краще, ніж конкуренти. Така стратегія може спиратися як на диференціацію, так і на лідерство по витратах, або і на те, і на інше, але тільки у рамках цільового сегменту. Проте низька ринкова доля у разі невдалої реалізації стратегії може істотно підірвати конкурентоспроможність компанії. Найважливішими здібностями, які повинна мати компанія, що приймає цю стратегію, є з генерування маркетингових новацій, здійснення продуктових новацій.

Дослідження свідчать, якщо позиціонування здійснюється більше ніж за трьома ознаками, то воно є неефективним, оскільки не відкладається у свідомості споживача.

Результатом виконання підрозділу стала узгоджена система рішень щодо ринкової поведінки стартап-компанії, яка визначає напрями роботи стартап-компанії на ринку.

Таблиця 4.16 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Функціонування програми в будь-який час, без помилок	Стратегія диференціації	Залучення великої бази даних, експериментів, що зробить прогнозування швидшим і точнішим	Дозволяє мати доступ в будь-який час до отриманих даних
2	Велика статистика даних по різних напрямках за різний час		Залучення в систему великої кількості показників, інтелектуальна система підбору відповідних значень до завдання користувача	Дозволяє працювати з перевіреними даними Дозволяє використовувати відкриті дані для створення прогнозу

4.5 Розроблення маркетингової програми стартап-проекту

Сформовано маркетингову концепцію товару, який отримає споживач. Для цього у таблиці 4.17 підсумовано результати попереднього аналізу конкурентоспроможності товару. Концепція товару - письмовий опис фізичних та інших характеристик товару, які сприймаються споживачем, і набору вигод, які він обіцяє певній групі споживачів.

Таблиця 4.17 – Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Швидке та доступне прогнозування	Велика кількість моделей та статистик, якими можна скористатися без зайвих пошуків	Висока точність роботи програми в будь-який час у будь-якому місці Простота реєстрації в системі

Розроблено тривірневу маркетингову модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (таблиця 4.19).

1-й рівень - при формуванні задуму товару вирішується питання щодо того, засобом вирішення якої потреби і / або проблеми буде даний товар, яка його основна вигода. Дане питання безпосередньо пов'язаний з формуванням технічного завдання в процесі розробки конструкторської документації на виріб.

2-й рівень - цей рівень являє рішення того, як буде реалізований товар в реальному/ включає в себе якість, властивості, дизайн, упаковку, ціну.

3-й рівень - товар з підкріпленням (супроводом) - додаткові послуги та переваги для споживача, що створюються на основі товару за задумом і товару в реальному виконанні.

Таблиця 4.18 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Додаток, що дозволяє швидко та зручно знайти моделювати та прогнозувати корупційну поведінку в даному середовищі		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. функція вибору кращої моделі		
	2. функція пошуку нових відкритих даних		
	Якість: за рахунок комбінування моделей		
	Пакування відсутнє		
	Марка: FinanceForYou		
III. Товар із підкріпленням	До продажу		
	Після продажу		
За рахунок чого потенційний товар буде захищено від копіювання: за рахунок закритості вихідного коду додатку, захист ідеї товару.			

Після формування маркетингової моделі товару слід відмітити, що проект буде захищено від копіювання за допомогою ноу-хау. Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (таблиця 4.19). Аналіз проведено експертним методом.

Таблиця 4.19 – Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	130-160 грн/день	120-240 грн/день	2000-4000грн/тиждень (новачки)	110-260 грн/день

Наступним кроком є визначення оптимальної системи збуту, в межах якого було прийняте рішення (таблиця 4.20): проводити збут власними силами і залучати сторонніх посередників, користуватися однорівневим каналом збуту.

Таблиця 4.20 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Завантажують додатки за рекомендаціями у соціальних мережах та інтернет-ресурсах	Зберігання, встановлення контакту, інформування.	Однорівневий	Залучена

Стратегія лідерства по витратах передбачає, що компанія за рахунок чинників внутрішнього і/або зовнішнього середовища може забезпечити більшу, ніж у конкурентів маржу між собівартістю товару і середньо-ринковою ціною (або ж ціною головного конкурента). Зокрема, ця стратегія припускає, що за рахунок великих можливостей по об'ємах збуту товарів (портфеля укладених контрактів на постачання) і продуктивності підприємство може добитися менших витрат. Ця стратегія зазвичай тісно пов'язана з можливістю досягнення ефекту масштабу і досвіду.

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (таблиця 4.21).

Таблиця 4.21 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Перед вирішенням задачі прогнозування клієнти шукають в Інтернеті корисні поради щодо сервісів, які можуть їм знадобитися, і завантажують їх згідно з рекомендаціями статей.	Соціальні мережі (facebook, twitter, linkedin)	Зручність та якість прогнозування, легкість у користуванні	Впевнити клієнта у тому, що FinanceForYou – найкращий спосіб прогнозування інструментів фінансового ринку	Повідомлення у соціальних мережах, короткі ролики

Результатом підрозділу стала ринкова (маркетингова) програма, що включає в себе концепції товару, збуту, просування та попередній аналіз можливостей ціноутворення, спирається на цінності та потреби потенційних клієнтів, конкурентні переваги ідеї, стан та динаміку ринкового середовища, в межах якого впроваджено проект, та відповідну обрану альтернативу ринкової поведінки.

4.6 Висновки до розділу

У даному розділі було проведено аналіз програмного продукту у якості стартап проекту. Можна зазначити що у проекті низьку можливість до комерціалізації, адже він в першу чергу зацікавить великі компанії, які займаються розробкою подібних приладів. Він не є універсальним, а тому може зацікавити лише як підхід до вирішення проблем зі сторонніми каналами, а не як готовий товар.

На ринку наявна монополістична конкуренція, існує декілька фірм-конкурентів, тому вихід на нього не буде легким. Проект є конкурентоспроможним лише завдяки, в першу чергу, своїй високій надійності. Також проект включає витрати на мікроконтролери, де ідея буде реалізована.

Для впровадження ринкової реалізації проекту слід обрати альтернативу, яка передбачає розробку програмного продукту, а потім якісну рекламу та PR, сконцентровану навколо позитивних характеристиках даного програмного продукту, таких як низька ціна, надійність, безпека і т.д.

З огляду на проведений аналіз, можна чітко сказати, що подальша імплементація проекту не є доцільною, адже він може не знайти свою цільову аудиторію та зайняти місце на ринку дуже багатьма зусиллями й скоріше не буде прибутковим.

ВИСНОВКИ

Був проведений огляд літератури для обробки і аналізу зображень саме у контексті проблеми розпізнавання рукописного тексту. Запропонована система реалізована як програмний продукт на мові Python.

Системи розпізнавання рукописного тексту побудовна для історичних документів, написаних на івриті. Бінаризація була застосована як перший крок. Потім знаходимо текстова область, тобто зображення змінені, так що є на даному етапі чорні літери чітко видно на білому фоні. Цей крок є досить значним, оскільки він впливає на етапі сегментації ліній. Наступним кроком є проекція гістограми для сегментації ліній. У дисертації цей підхід до сегментації ліній модифікується пошуком максимумів у гістограмі. Далі розглянуті методи для реконструкції літер у деградованих історичних документів. На цьому кроці запропонована система також вирішує проблему шуму. Наш підхід передбачає одночасне розпізнавання символів та сегментацію, вирішуючи проблему перекриваючих символів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Maad M. M. Handwriting Recognition Methods / Maad M. M. // IEEE Transactions on Signal Processing. - 2005. - pp. 1-3.
2. Fischer A. Character prototype selection for handwriting recognition in historical documents with graph similarity features/ Fischer A., Bunke H. // Proc. 19th European Signal Processing Conference. - 2011. - pp. 1435–1439.
3. Gatos B. An adaptive binarisation technique for low quality historical documents. Document analysis systems. / Gatos B., Pratikakis I., Perantonis S. J. – Berlin: VI Lecture notes in computer science. - 2004. – Vol. 3163, pp. 102–113.
4. Thillou C., Color binarization for complex camera-based images / Thillou C., Gosselin B. // Electronic Imaging Conference of the International Society for Optical Imaging. - 2005. – pp. 1 – 8.
5. Otsu, N. A threshold selection method from grey-level histograms. / Otsu, N. // IEEE Transactions on System, Man, and Cybernetics. - 1979. – pp. 62-66.
6. Герасимов Б. М. Человеко-машинные системы принятия решений с элементами искусственного интеллекта / Герасимов Б. М. , Тарасов В. А., Токарев И. Б. – К.: Наукова Думка, 1993. – 184 с.
7. Химмельблау Д. Анализ процессов статистическими методами / Химмельблау Д. – М.: Мир, 1973. – 957 с
8. Shafer G. Probabilistic Expert Systems. [Text] / Society for Industrial and Applied Mathematics, Philadelphia, PA, 1996.
9. Lauritzen S. L. and Spiegelhalter D. J. Local computations with probabilities on graphical structures and their application to expert systems [Text] / Journal of Royal Statistics Society, series B – statistical methodology. – 1988. – Vol. 50, No2. – 157-194 p.
10. Lauritzen S. L. and Jensen F. V. Local computation with valuations from a commutative semigroup [Text] / Annals of Mathematics and Artificial Intelligence, – New York: Springer, 1997. – Vol. 21, No 1. – pp. 51-69.

11. Хабаров С. П. Экспертные системы [Электронный ресурс] : Конспект лекций. – Режим доступа : <http://firm.trade.spb.ru/serp>.
12. Casey, R. A survey of methods and strategies in character segmentation./ Casey R., Lecolinet, E.: // IEEE Transactions on System, 1996 - pp. 690–706.

ДОДАТОК А ЛІСТИНГ ПРОГРАМИ

```

robot = Robot(sigma, delta, model_parameters["init_robot_pos"],
actual_angles=model_parameters["robot_actual_angles"],
               compass_angles=model_parameters["robot_compass_angles"])
result = run_model(robot, world, sigma, delta,
init_particles_pos=model_parameters["init_particles_pos"],
filtering_algorithm=algorithm)
error_norms = list(map(lambda x: np.linalg.norm(np.subtract(x[0], x[1])),
result))
plt.plot(error_norms)
plt.savefig(algorithm + ".png")

import cv2
from matplotlib import pyplot as plt
import os
import numpy as np
import peakutils

# from skimage import measure

# from PIL import Image, ImageFont, ImageDraw

# Folder with original images
pathIn =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/ImageData/"

# Folders for output
pathOut1 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputOTSU/"
pathOut2 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputGAU/"
pathOut3 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputMEAN/"
pathOut4 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputCropped/"
pathOut5 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputCroppedGAU/"
pathOut6 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/OutputCroppedSmall/"
pathOut7 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/Lines/"
pathOut8 =
"/Users/olgamulava/Documents/Uni/5Groningen/HandwritingRecognition/Code/Lines/ConnectedComponents/"

# For output in console
np.set_printoptions(threshold=np.inf)

# def largest_component(image):
#     # image must be already binarized

```

```

#     nb_components, labels, stats, centroids =
cv2.connectedComponentsWithStats(image, connectivity=8, ltype=cv2.CV_32S)
#     # nb_components is the number of labels
#     # Labels is a matrix the size of the input image where each element has a
value equal to its label.
#     # Stats is the stat matrix - the most important
#     # Centroids is a matrix with the x and y locations of each centroid. The
row in this matrix corresponds to the label number.
#
#     sizes = stats[:, -1]
#
#     max_label = 1
#     max_size = sizes[1]
#     for i in range(2, nb_components):
#         if sizes[i] > max_size:
#             max_label = i
#             max_size = sizes[i]
#
#     # Crop just connected component with background
#     # img2 = np.zeros(labels.shape)
#     # img2[labels == max_label] = 255
#
#     # Crop image[y1:y2, x1:x2]
#     img2 = image[stats[max_label, cv2.CC_STAT_TOP]:stats[max_label,
cv2.CC_STAT_TOP]+stats[max_label, cv2.CC_STAT_HEIGHT], stats[max_label,
cv2.CC_STAT_LEFT]:stats[max_label, cv2.CC_STAT_LEFT]+stats[max_label,
cv2.CC_STAT_WIDTH]]
#
#     # Biggest component
#     return img2

def largest_contour(img):
    img2 = img.copy()
    # Parameters: first one is source image, second is contour retrieval mode,
third is contour approximation method
    _, contours, hierarchy = cv2.findContours(img2, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)

    # Find the index of the largest contour
    areas = [cv2.contourArea(c) for c in contours]
    max_index = np.argmax(areas)
    cnt = contours[max_index]

    # Convex Hull
    # hull = cv2.convexHull(cnt)#, returnPoints=False)

    # cv2.drawContours(img2, [hull], -1, 50, 1)

    # Mask initialized as white image
    mask = np.ones(img2.shape[:2], dtype="uint8") * 255

    # Draw on mask image filled biggest contour
    cv2.drawContours(mask, [cnt], -1, 0, -1) # [hull]

    # cv2.drawContours(img2, contours, -1, (255, 255, 0), -1)
    # cv2.imwrite('TestCharacter.png', img2)

    # Out initialized as white image
    out = np.ones(img2.shape[:2], dtype="uint8") * 255
    out[mask == 0] = img2[mask == 0]

    # Find coordinates of biggest contour cnt
    x, y, w, h = cv2.boundingRect(cnt)

```

```

# Crop image[y1:y2, x1:x2]
# imgReturn = out[y:y + h, x:x + w]

return out # imgReturn # np.array(out)

# # Character Segmentation
# def char_contours(img):
#     # Image must be already binarized
#     # Copy the input image
#     # returnImg = img.copy()
#     # # Histogram for Line
#     # img_row_sum = np.sum(returnImg, axis=0) # axis=0 - x coordinate
#     # plt.figure(0)
#     # plt.plot(img_row_sum)
#     # plt.savefig('HistogramForLine.png', dpi=500)
#     returnImg = np.asarray(img) # change
#     _, contours, hierarchy = cv2.findContours(returnImg, mode=cv2.RETR_LIST,
method=cv2.CHAIN_APPROX_NONE)
#     cv2.drawContours(returnImg, contours, -1, 100, 3) # last parameter is
thickness = 3
#     return np.array(returnImg)

# Line segmentation function returns an array of lines images
def lines_segmentation(img_input):
    img_copy = img_input.copy()
    img = np.invert(img_input)

    # Make a graph from sum of pixels in each row
    img_row_sum = np.sum(img, axis=1).tolist() # axis=1 - y coordinate

    peaks = peakutils.indexes(np.array(img_row_sum), thres=0.15, min_dist=60) #
thres=0.02 / max(cb), min_dist=100)
    # thres - Only the peaks with amplitude higher than the threshold will be
detected
    # min_dist - Minimum distance between each detected peak

    print(peaks)

    plt.title('Histogram')
    # plt.hist(indexes, bins='auto', orientation='horizontal')
    plt.plot(img_row_sum)
    plt.savefig('Histogram.png', dpi=500)

    # Find average line height
    list_avg = []
    for i in range(len(peaks) - 1):
        list_avg.append((peaks[i+1] - peaks[i]))
    avg_height = int(sum(list_avg) / float(len(list_avg)))
    print(avg_height)

    H = img.shape[0]
    lines = [] # np.zeros(len(peaks) + 1)
    lines.append(peaks[0] - avg_height / 2) # first line
    for peak in peaks:

```

```

        lines.append(min(peak + avg_height / 2, H))

# Converting float type to int
lines = np.array(lines)
lines = lines.astype(int)
# print(lines)

# th = sum(img_row_sum) / float(len(img_row_sum))
# H, W = img.shape[:2]
#
# lowers = np.array([y for y in range(H - 1) if img_row_sum[y] <= th and
img_row_sum[y + 1] > th])
# uppers = np.array([y for y in range(H - 1) if img_row_sum[y] > th and
img_row_sum[y + 1] <= th])
#
# lines = np.zeros(len(uppers))
# lines[0] = max(uppers[0] - 20, 0)
# lines[-1] = min(lowers[-1] + 20, H)
#
# for i in range(1, len(uppers) - 1):
#     lines[i] = (lowers[i - 1] + uppers[i]) / 2

# # Converting float type to int
# lines = lines.astype(int)

# print(lines)
# for y in uppers:
#     cv2.line(newImage, (0, y), (W, y), (255, 0, 0), 1)
#
# for y in lowers:
#     cv2.line(newImage, (0, y), (W, y), (255, 0, 0), 1)

# Width of the image
W = img.shape[1]

# for y in lines: # lines:
#     cv2.line(img_copy, (0, y), (W, y), 0, 1) # 0 - black
#
# cv2.imwrite("LineSegmentation.png", img_copy)

lines_array = []

for i in range(len(lines) - 1):
    # To crop image[y1:y2, x1:x2]
    img_line = np.array(img_copy[lines[i]:lines[i + 1], 0:W])
    lines_array.append(img_line)

return lines_array

k = 0

def get_connected_components(imgline):
    global k
    k = k + 1
    imgreturn = imgline.copy()
    # image must be already binarized
    nb_components, labels, stats, centroids =
cv2.connectedComponentsWithStats(imgline, connectivity=8, ltype=cv2.CV_32S)
    # charCandidates = np.ones(imgline.shape, dtype="uint8") * 255
    print(nb_components)

```

```

# List of all components images from the line image
conComponentsArray = []

# Use CC_STAT_AREA to filter?

for label in range(nb_components):
    # Retrieving the width of the bounding box of the component
    width = stats[label, cv2.CC_STAT_WIDTH]

    # Retrieving the height of the bounding box of the component
    height = stats[label, cv2.CC_STAT_HEIGHT]

    # Retrieving the leftmost coordinate of the bounding box of the
component
    x = stats[label, cv2.CC_STAT_LEFT]

    # retrieving the topmost coordinate of the bounding box of the component
    y = stats[label, cv2.CC_STAT_TOP]

    # creating the image of the component
    component = imgline[y:y + height, x:x + width]

    conComponentsArray.append(component)

    cv2.rectangle(imgreturn, (x, y), (x + width, y + height), color=0,
thickness=2)

cv2.imwrite("cc" + str(k) + ".png", imgreturn)

print("len(conComponentsArray) =", len(conComponentsArray))

return conComponentsArray

drawing = imgline.copy()

for compLabel in range(1, nb_components, 1):
    cv2.rectangle(drawing, (stats[compLabel, 0], stats[compLabel, 1]),
(stats[compLabel, 0] + stats[compLabel, 2], stats[compLabel, 1] +
stats[compLabel, 3]), (0, 0, 255), 2)
    cv2.imwrite(pathOut8 + "DrawConComp" + str(imgline) + ".png", drawing)

sizes = stats[:, -1]

max_label = 1
max_size = sizes[1]
for i in range(2, nb_components):
    if sizes[i] > max_size:
        max_label = i
        max_size = sizes[i]

# Crop just connected component with background
img2 = np.zeros(labels.shape)
img2[labels == max_label] = 255

# Crop image[y1:y2, x1:x2]
img2 = imgline[stats[max_label, cv2.CC_STAT_TOP]:stats[max_label,
cv2.CC_STAT_TOP]+stats[max_label, cv2.CC_STAT_HEIGHT], stats[max_label,

```

```

cv2.CC_STAT_LEFT]:stats[max_label, cv2.CC_STAT_LEFT]+stats[max_label,
cv2.CC_STAT_WIDTH]]

    # Biggest component - img2
    conComponentsArray.append(img2)

# Draw contours on a line image
def draw_contours(imgline):
    global k
    k = k + 1

    # imgLineCopy = cv2.subtract(255, imgline)

    # imgLineCopy = imgline.copy()

    # Add white borders on each side
    bordersize = 1
    imgWithBorder = cv2.copyMakeBorder(imgline, top=bordersize,
bottom=bordersize, left=bordersize, right=bordersize,
borderType=cv2.BORDER_CONSTANT, value=[255, 255, 255])
    cv2.imwrite("TestBorder.png", imgWithBorder)

    # Parameters: first one is source image, second is contour retrieval mode,
    third is contour approximation method
    _, contours, hierarchy = cv2.findContours(imgWithBorder, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)

    contoursArray = []

    for c in contours:
        if 50 <= cv2.contourArea(c) <= 10000:
            contoursArray.append(c)

    # Find areas
    # areas = [cv2.contourArea(c) for c in contours]
    # print("areas", areas)

    # ? Why black line ? - Fix that
    img_return = cv2.drawContours(imgWithBorder, contoursArray, -1, color=0,
thickness=2)

    # Crop borders
    w, h = img_return.shape[:2]
    img_return = img_return[bordersize:w, bordersize:h-bordersize]

    cv2.imwrite("TestContoursWords" + str(k) + ".png", imgWithBorder)

    # Return the same image with contours drawn on the top
    return img_return
    # return contoursArray

# Read an image
img = cv2.imread(pathIn + "P632-Fg002-R-C01-R01-fused.jpg", 0) # 0 - grayscale
# img = cv2.imread(pathIn + "P423-1-Fg002-R-C02-R01-fused.jpg", 0) # 0 -
grayscale
# img = cv2.imread(pathIn + "124-Fg004.pgm", 0) # 0 - grayscale

# # Invert colors black to white, white to black
# imgtest = np.invert(img)

```

```

# Show
# cv2.imshow('My image', img)
# cv2.waitKey()

ret, imgBin = cv2.threshold(img, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)

goodImage = largest_contour(imgBin)

# goodImage1 = largest_contour(imgBin)
# contours = get_contours(goodImage1)
# goodImage = cv2.drawContours(goodImage1, contours, -1, color=0, thickness=1)

cv2.imwrite('test.png', goodImage)

# Worse performance
# cropImage = largest_component(imgBin)
# cv2.imwrite('cropTest.png', cropImage)
# goodImage = largest_contour(cropImage)
# cv2.imwrite('test.png', goodImage)

lines_imgs = lines_segmentation(goodImage)
lines_imgs_contours = []

for i in range(len(lines_imgs)):
    # cv2.imwrite(pathOut7 + "SimpleTest" + "LineImage" + str(i) + ".png",
    lines_imgs[i])

    # CONTOURS
    line_contoured = draw_contours(lines_imgs[i]) # lines_imgs[i]
    # cv2.imwrite(pathOut7 + "Simple" + "LineImage" + str(i) + ".png",
    line_contoured)
    lines_imgs_contours.append(line_contoured)

    # for j in range(len(contours)):
    #     cv2.imwrite(pathOut8 + "test" + "LineImage" + str(i) + "CONTOUR" +
    str(j) + ".png", contours[j])

    # CONNECTED COMPONENTS ?
    # conComponents = get_connected_components(line_i)
    # # # print(conComponents)
    # for j in range(len(conComponents)):
    #     cv2.imwrite(pathOut8 + "test" + "LineImage" + str(i) + "OLYA" +
    str(j) + ".png", conComponents[j])

# cv2.imwrite("Line Image", goodImage[20:100, 20:100])
# cv2.imwrite("LineImage.png", Image.fromarray(goodImage[1:40, 1:30]))
# Image.fromarray(goodImage[0:200, 0:200]).save("LineImage.png")

# Open the folder with input images
listing = os.listdir(pathIn)

for file in listing:
    if not file.startswith('.'): # check for hidden files

        # Read an image
        img = cv2.imread(pathIn + file, 0)

        # OTSU Binarization

```



```

ret, imgi = cv2.threshold(img, 0, 255, cv2.THRESH_BINARY +
cv2.THRESH_OTSU)
cv2.imwrite(pathOut1 + file, imgi)

# Cropped important part as a largest connected component with borders,
small size
goodImage = largest_contour(imgi)
cv2.imwrite(pathOut6 + file, goodImage)

# cv2.imwrite(pathOut7 + file, np.array(char_contours(goodImage)))

# # GAUSSIAN Binarization
# imgi = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C,
cv2.THRESH_BINARY, 11, 2)
# cv2.imwrite(pathOut2 + file, imgi)
#
# # reverse black to white to make this work
# # cv2.imwrite(pathOut5 + file, largest_component(imgi))
#
# # MEAN Binarization
# imgi = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_MEAN_C,
cv2.THRESH_BINARY, 11, 2)
# cv2.imwrite(pathOut3 + file, imgi)

# Creat a list of segmented lines
lines_imgs = lines_segmentation(goodImage)

# List of contoured line images
lines_imgs_contours = []
for i in range(len(lines_imgs)):
    # Draw contours
    # line_contoured = draw_contours(lines_imgs[i]) # lines_imgs[i]
    cv2.imwrite(pathOut7 + file + "LineImage" + str(i) + ".png",
line_contoured) # could be commented later
    # Append a contoured line image
    lines_imgs_contours.append(line_contoured)

image1 = cv2.imread(pathOut1 + "P344-Fg001-R-C01-R01-fused.jpg", 0)
# cv2.imshow('My image', image1)
# cv2.waitKey()

# Template is one char
template = cv2.imread("CorrectData/" + "Bet.png", 0)
w, h = template.shape[::-1]

# finding the location of a template image in a larger image
res = cv2.matchTemplate(image1, template, cv2.TM_CCORR_NORMED)

# n
min_val, max_val, min_loc, max_loc = cv2.minMaxLoc(res)

top_left = max_loc
bottom_right = (top_left[0] + w, top_left[1] + h)

cv2.rectangle(image1, top_left, bottom_right, (100, 100, 50), 2)
cv2.imwrite("my.png", image1)
# cv2.imshow("TM_CCORR_NORMED", image1)
cv2.waitKey()

```

```
print(top_left, bottom_right)

# plt.subplot(121)
# plt.imshow(res, cmap='gray')
plt.xticks([])
plt.yticks([])
# plt.subplot(122)
plt.imshow(image1, cmap='gray')
plt.show()
```