

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»  
Кафедра інформатики факультету інформатики

## **ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ПОМИЛОК В ДОКУМЕНТАХ**

**Текстова частина до курсової роботи  
за спеціальністю “Комп’ютерні Науки” 122**

Керівник курсової роботи  
доц. Ігнатенко О.П.

\_\_\_\_\_  
(підпис)  
“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

Виконала студентка  
Жиркова А.П.  
“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

Київ 2020

ЗМІСТ	Стор
Анотація	3
<b>ВСТУП</b>	4
<b>РОЗДІЛ 1: ОСОБЛИВОСТІ ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОБОТИ З ДАНИМИ, ПРЕДСТАВЛЕНИМИ У ВИГЛЯДІ ЗОБРАЖЕНЬ</b>	7
1.1 Робота з зображеннями.	7
1.1.1 Представлення зображень.	7
1.1.2 Обробка зображень.	9
1.2 Використання машинного навчання для задач, пов'язаних з обробкою зображень.	14
<b>РОЗДІЛ 2: ЗАСТОСУВАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ПОМИЛОК У ДОКУМЕНТАХ</b>	21
2.1 Постановка задачі	21
2.2 Існуючі методи	22
2.3 Імплементация методу	24
2.3.1 Завантаження даних	24
2.3.2 Попередня обробка документів	27
2.3.3 Класифікація документів за наявністю печатки	31
2.4 Результати	33
2.4.1 Залежність точності моделі від кількості навчальної вибірки	33
<b>Висновки</b>	38
Список використаних джерел	40
Додатки	42

## **Анотація**

**Тема курсової роботи:** Використання машинного навчання для розпізнавання помилок в документах

**Студентка** Жиркова Анастасія Павлівна

**Рік навчання, спеціальність, факультет:** 1-й рік навчання (магістерська програма), комп'ютерні науки, факультет інформатики

**Науковий керівник** доцент Ігнатенко О.П.

*Мета роботи полягає в аналізі методів машинного навчання для розпізнавання помилок в документах, методів роботи з зображеннями та їх класифікації, для чого на прикладі реальної задачі було оброблено більше 4000 документів, представлених у вигляді зображень, та класифіковано за наявністю або відсутністю печатки.*

**Ключові слова:** машинне навчання, згорткові нейронні мережі, класифікація, обробка зображень, веб-скрапінг.

## Вступ

Машинне навчання - вже не новий, але дуже популярний напрям для досліджень та розробки, орієнтований на роботу з різними видами даних, розуміння їх структури та взаємозв'язків.

Об'єм даних, представлений у вигляді зображень, значно виріс з розвитком технологій та популяризацією фотографій як способу поділитися певною інформацією. Присутність камери на всіх мобільних пристроях, що випускаються останнім часом, та поліпшення якості фотографій, є значним рушієм для поширення використання даних такого виду. А отже, з'являються і методи обробки подібної інформації, де центральним поняттям є термін “computer vision”, який на українську переводиться як “комп'ютерний зір” та означає процес обробки графічних даних, який має на меті, наприклад, розпізнавання об'єктів певного класу на зображеннях.

Можливість сканувати або фотографувати документи дозволяють зберігати їх у вигляді зображень, а отже, і застосовувати до них відповідні методи обробки. Наявність або відсутність печаток на зображенні є типовим представником задачі класифікації, розв'язати яку можливо за допомогою машинного навчання.

**Актуальність теми.** При складанні документів виникають помилки різного типу, які важко відстежити та навіть при дуже уважному перегляді можна пропустити. Тоді виникає потреба обробляти документи в автоматизованому режимі, щоб відслідковувати помилки та мінімізувати їхню присутність у документах. Використання машинного навчання для розв'язання задач цього класу є актуальним способом вирішення подібних проблем.

**Мета і завдання дослідження.** Метою курсової роботи є розробка моделі згорткової нейронної мережі для коректної класифікації документів за

наявністю або відсутністю печаток на ньому. Для досягнення поставленої мети вирішуються наступні *завдання*:

- Вивчаються різні методи роботи з зображеннями;
- Досліджується можливість використання машинного навчання для роботи з зображеннями;
- Проводиться аналіз існуючих методів вирішення поставленої задачі;
- Проводиться збір та обробка даних;
- Будується нейронна мережа для класифікації документів;
- Вивчається залежність кількості даних для навчання моделі на точність класифікації.

*Об'єктом дослідження* є методи машинного навчання для роботи з зображеннями.

*Предметом дослідження* є пошук оптимальної моделі машинного навчання для класифікації зображень за наявністю або відсутністю об'єктів певного класу (а саме, печаток).

**Методи дослідження.** В дослідженні використані методи добування даних з мережі Інтернет, методи обробки даних для їхнього подальшого використання алгоритмами машинного навчання та методи побудови згорткових нейронних мереж для класифікації зображень.

**Наукова новизна отриманих результатів.** В результаті виконання курсової роботи було:

*вперше:*

- Проведено попередню обробку завантажених документів для подальшої класифікації;
- Побудовано модель нейронної мережі для класифікації документів за наявністю або відсутністю печаток;

*удосконалено:*

- Методи веб-скрапінгу для добування документів з веб-сайту Prozorro;

*набули подальшого розвитку:*

- Методи машинного навчання для розв'язання задачі класифікації документів за наявністю або відсутністю печатки.

**Практичне значення отриманих результатів.** Відсутність печаток на документі є грубою помилкою при складанні різних типів угод, відомостей і т.д. Структури, що працюють з документами, повинні перевіряти дотримання цих документів усім правилам, автоматизація ж даного процесу може спростити їхню роботу.

# **РОЗДІЛ 1: ОСОБЛИВОСТІ ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОБОТИ З ДАНИМИ, ПРЕДСТАВЛЕНИМИ У ВИГЛЯДІ ЗОБРАЖЕНЬ**

## **1.1 Робота з зображеннями.**

Останнім часом все більший відсоток електронної інформації представлений саме графічною інформацією. Тому робота з зображеннями займає важливу позицію в сфері Data Science, яка в свою чергу займається аналізом і обробкою інформації.

### **1.1.1 Представлення зображень.**

Графічне зображення - це функція  $f(x, y)$ , яка є проекцією 3-вимірної сцени у 2-вимірну, де  $x, y$  вказують на розташування пікселя та містять інформацію про його колір. Математично, це матриця двовимірного зображення [1]. Кожний елемент цієї матриці містить значення, яке відповідає за колір.

Існує декілька кольорових моделей, які широко використовуються для представлення графічної інформації в електронному вигляді.

Найпростіша кольорова модель - це зображення у градації сірого (від англ. “grayscale”). Кожний відтінок даної моделі представлений у діапазоні від 0 до 255. Зазвичай, 0 позначає повністю чорний, а 255 - повністю білий. Проміжні значення позначають відтінки, які утворюються у результаті накладання даних кольорів один на другий.

Наступна кольорова модель - RGB, що розшифровується як “червоний, зелений, синій” (від англ. “red, green, blue”). Вона використовує ці три

кольори, які є основними і які для відображення певного відтінку накладаються один на другий. Варто помітити, що процес накладання кольорів відбувається на чорному фоні. В даній моделі для позначення певного кольору необхідні значення інтенсивності кожного з кольорів, що присутні у назві даної кольорової моделі.

Також відома кольорова модель - CMYK, яка є розширенням моделі CMY. Вона використовує блакитний, фіолетовий, жовтий та чорний кольори, останній з яких утворюється за допомогою накладання один на інший усіх попередніх (від англ. “cyan, magenta, yellow, key”). Дана модель характеризується меншою кількістю можливих кольорів, ніж у попередній моделі, та широко використовується для друку. Тому можливі казуси, коли кольори на роздрукованому зображенні не відповідають тим, що відображаються на електронних пристроях.

Також існує модель HSI, яка представляє колір як сукупність відтінку, насиченості та інтенсивності (від англ. “hue, saturation, intensity”). Вона краще підходить для розробки алгоритмів, що обробляють зображення, через те, що модель RGB у порівнянні з даною, є більш обмеженою, а отже, має менше можливостей для опису кольорів [2].

Описані вище моделі надають можливість працювати з зображеннями алгоритмам машинного навчання, оскільки усі зображення є сукупністю числових значень - типом даних, з якими вони вміють працювати.



### 1.1.2 Обробка зображень.

Для покращення якості роботи різних моделей машинного навчання, до графічної інформації застосовуються деякі методи обробки зображень. Вони допомагають отримати більш точний результат. Серед таких методів можна виділити видалення шумів, бінаризацію зображення, сегментацію та згладжування.

Отже, бінаризація зображення - це процес перетворення чорно-білого або кольорового зображення в зображення, яке має тільки два значення кольорів: 0 (чорний) та 1 (білий). Для можливості перегляду зображення, 1 замінюється на 255. Бінаризація використовує правило: пікселі, які мають значення менше заданого порогу, перетворюються на нуль, а ті, значення яких більше або рівне цьому порогу, стають одиницями. Ефективність роботи даного методу дуже залежить від складності зображень: на більш простих зображеннях він дає кращий результат, ніж на більш складних [3].

Дуже важливою темою в обробці зображень є присутність різних видів шумів на зображеннях та методи їх зменшення. Отже, шум - це випадкова зміна інтенсивності зображення, яка відображається у вигляді зерен на ньому. Шуми можуть утворитися під час зйомки або під час передачі зображення. В електронному представленні це означає, що пікселі в зображенні замість істинних значень мають якісь хибні. Процес видалення шумів полягає в зміні таких аномальних значень на ті, що близькі до істинних. Загалом, алгоритми видалення шумів полягають у зміні значень пікселів, які радикально відрізняються від сусідніх, шляхом наближення їх до тих, що знаходяться поруч. Цей процес можна назвати процесом згладжування зображення, який оброблює всі області окрім тих, що мають контрастні переходи (межі).

За типами шумів розрізняють наступні: Гаусовий; імпульсний, або ж “сіль-та-перець” (від англ. “salt-and-pepper”); зернистість та інші [4]. Звісно, усі вони мають різні характеристики, а отже, і підходи до їх видалення мають бути різними.

Стандартна модель Гаусового шуму є адитивною - залежить від кожного пікселя та від інтенсивності сигналу. Вона є ідеалізованою формою білого шуму, викликаним випадковими коливаннями в сигналі. Цей тип шуму є основним, та локалізується на темних областях зображення. Гістограма, яка ілюструє графік залежності величини викривлення значення пікселя від частоти, з якою це відбувається, зображує нормальний розподіл шуму. Існує теорема, яка говорить про те, що сума різних шумів має тенденцію наближуватися до Гаусового розподілу.

На рисунках нижче представлені оригінальне зображення у градації сірого та воно ж з Гаусовим шумом.



Рис. 1.1 - Оригінальне зображення [13]



Рис. 1.2 - Зображення з доданим Гаусовим шумом

Шум “сіль-та-перець” часто називають імпульсним, або випадковим, або незалежним. Тут пікселі з хибним значенням мають значення, які дуже сильно відрізняються за кольором від справжніх. Такий вид шуму може бути спричинений раптовим та дуже різким порушенням сигналу. Частіше за все, при псуванні зображення даним видом шуму, його зазнають порівняно мала кількість пікселів.



Рис. 1.3 - Зображення з доданим шумом типу “сіль-та-перець”

Незалежно від того, бачить людина шум чи ні, він має місце бути на зображенні. Тому прибирання шуму є бажаним кроком при роботі з зображеннями.

Для того, щоб позбутися його, існує багато різних алгоритмів, наприклад, лінійне згладжування, медіанний фільтр та інші. В таких алгоритмах, три основні кольори (червоний, зелений та синій) обробляються окремо. В кінці роботи яких вони поєднуються для формування кольорових зображень.

Лінійне згладжування використовується для зменшення деяких типів шумів. Воно має тенденцію руйнувати лінію та дрібні деталі зображення, а також погано працювати з шумами, які залежать від сигналу. Працює алгоритм наступним чином: для кожного пікселю береться середнє або середньозважене значення самого пікселю та його найближчих сусідів. На

практиці, для видалення шумів даний алгоритм видає не найкращі результати, а тому використовується дуже нечасто.

Медіанний фільтр працює так: береться шматок матриці розміром  $3 \times 3$ , усі значення пікселів записуються від найменшого до найбільшого і обирається медіанне значення зі списку (є серединою даного списку). Отриманим значенням замінюється центральний елемент поточної матриці. Таким чином повністю обробляється зображення. Медіанний фільтр, як і попередній, також не дуже добре працює з шумами, які залежать від сигналу, а тому для отримання кращих результатів було розроблено різні варіації цього алгоритму.

Медіанний фільтр гарно працює з видаленням таких типів шумів як “сіль-та-перець”, тобто, для випадкових шумів. Для видалення Гаусового шуму найкраще підходить фільтр Вінера [4].

Також важливим видом обробки зображень є сегментація зображення. Даний термін описує процес розбиття зображення на декілька регіонів або частин. Цей процес відбувається в залежності від пікселів, а точніше від схожості або відмінності їх характеристик. Прикладом задач сегментації можуть бути кластеризація регіонів пікселів в залежності від їх кольору та відокремлення заднього фону від того, що знаходиться на передньому. Метою розбиття зображення на сегменти може бути зацікавленість в опрацюванні певного шматка замість цілого зображення.

Згадана вище бінаризація зображення є одним з видів сегментації. Іншим видом сегментації є розпізнавання якихось схожостей на зображенні та утворення регіонів. Тут можна згадати такі методи як визначення регіонів та кластеризацію [5].

## **1.2 Використання машинного навчання для задач, пов'язаних з обробкою зображень.**

Використання машинного навчання, а зокрема штучних нейронних мереж, для задач, які обробляють зображення з певною метою (наприклад, для того, щоб класифікувати їх за певним параметром), є дуже популярним способом вирішення подібних проблем. Одним з найпопулярніших способів розв'язання даних задач є використання згорткових нейронних мереж (від англ. “Convolutional Neural Networks”, або “CNN”). Цей алгоритм є найефективнішим для роботи з зображеннями, оскільки його відмінністю є здатність розпізнавати важливість різних аспектів або об'єктів зображення та відрізняти їх між собою. Архітектура згорткових мереж подібна тому, як працює людський мозок, та була розроблена на прикладі організації частини людського мозку, яка відповідає за зорове сприйняття, а саме зорова кора: окремі нейрони реагують на подразнення тільки в обмеженій області поля зору, а поєднання роботи усіх нейронів покриває усе поле зору, що забезпечує сприйняття картини в цілому.

Згорткові мережі можуть чудово розпізнавати просторові залежності. Архітектура мережі забезпечує можливість повторного використання ваг та зменшує кількість задіяних параметрів, що дозволяє краще підлаштовуватися під дані, отримані з зображення. Простішими словами, мережу можна навчити розуміти складність зображення.

Вхідне зображення має відповідати певній кольоровій моделі. Якщо це RGB, то зображення представляється у вигляді матриць трьох каналів (червоного, зеленого та синього), кожна з яких відповідає за інтенсивність відповідного кольору на зображенні.

24	5	18
7	0	48
10	56	34

3	61	88
45	12	8
39	55	14

18	10	22
14	42	13
54	48	65

Рис. 1.4 - Матриці каналів для зображення у RGB розміром 3x3, які накладаються один на інший для відображення зображення

Якщо згадати стандартні розміри сучасних зображень (їх роздільна здатність), то не важко уявити, якою високою має бути потужність комп'ютера, необхідна для його обробки. Метою згорткових нейронних мереж є зменшення розмірності зображення для легшої обробки, без втрати характеристик зображення, які мають велике значення для передбачень.

Операція згортки зводиться до зменшення розмірності зображення шляхом обробки усієї матриці, при якій так зване ядро згортки, розмірністю, наприклад, 2x2, проходить по заданій матриці з якимось кроком (зазвичай, 1), та виконує певні обчислення (рахує, чому дорівнює сума усіх елементів, які обробляються ядром, помножені на певні ваги). Якщо ваги ядра дорівнюють

$((0, 1), (1, 0))$ , де 0 та 1 - елементи першого рядочку матриці, а 1 та 0 - елементи другого. Тоді застосовуючи операцію згортки, отримаємо:

24	64
96	61

Рис. 1.5 - Матриця каналу червоного кольору після операції згортки

Існує багато різних можливостей для згортки шляхом регулювання розміру ядра згортки та кроку. Якщо крок буде рівним 2, то кожний крок ядра в рамках даної операції згортки буде відрізнятися від попереднього на два елементи, а точніше, при згортанні матриці, ядро буде рухатися через один елемент матриці.

Існує декілька типів операції згортки: направлений на зменшення розмірності зображення та на збільшення (можливий варіант, коли розмірність зображення залишається незмінною).

Метою згортки є вилучення деяких важливих характеристик (наприклад, меж) з вхідного зображення. Зазвичай, перший шар згорткової нейронної мережі направлений на вилучення низькорівневих характеристик, таких як колір, напрямок градієнту, межі. Наступні шари відповідають за розуміння високорівневих функцій, що дає можливість аналізувати зображення в цілому.

Окрім згорткового шару, існує також об'єднуючий (від англ. “pooling layer”). Він на меті має також зменшення розмірності зображення, що дозволяє нейронній мережі швидше навчатися. Він є корисним і для витягування важливих характеристик з зображення, що забезпечує більш ефективне навчання моделі.



Існує два методи, що застосовуються для зменшення розмірності на цьому кроці: так звані Max Pooling та Average Pooling. В обох випадках використовується ядро для проходження усієї матриці, як і при виконанні згорткового шару. Перший метод при проходженні ядра повертає максимальне значення зі шматка матриці, до якої застосовується ядро. При другому методі - повертається середнє значення. Перший метод вважається кращим, оскільки він також грає роль пригнічувача шуму, тобто, при обході матриці, він, окрім зменшення її розмірності, ще й прибирає шуми з зображення.

Якщо за згортковим шаром застосовується об'єднуючий, то вони утворюють один шар згорткової нейронної мережі.

Після роботи згорткової нейронної мережі, для отримання цільової змінної, наприклад, у задачах класифікації, результат перетворюється з матриці у впорядкований список значень (тобто, кожне місце списку відводиться на елемент матриці, що розташований на певній позиції), після чого застосовуються звичайні нейронні мережі [6].

Нейронні мережі зазвичай мають декілька шарів, кожний з яких використовує певну функцію для підрахунків, передає вихідні значення у наступний шар, і так до тих пір, поки не отримається остаточний результат у останньому шарі. Перший шар називається вхідним (від англ. “input layer”), подальші - прихованими (від англ. “hidden layers”), та останній - вихідний (від англ. “output layer”). Нейронна мережа називається повністю-з'єднаною, якщо кожен нейрон кожного шару з'єднується (має зв'язок) з кожним нейроном наступного шару.

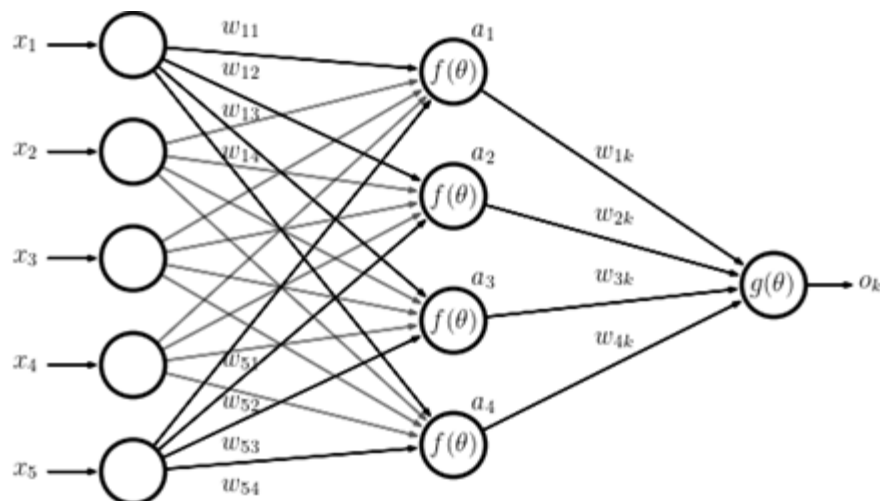


Рис 1.6 - Схема нейронної мережі [7]

На малюнку вище можна побачити, що кожний зв'язок між нейронами має певну вагу. Ця вага використовується для навчання, і з кожною ітерацією може мінятися, щоб краще відповідати моделі.

Ітерації потрібні, щоб проаналізувати ефективність попередньої моделі і шляхом підбору інших ваг покращити результат тренування.

Кожний шар має свою активаційну функцію. Вона обирається для обрахунку в кожному нейроні вихідного значення. Різні функції дають різний результат, а отже, існує більше можливостей для налаштування моделі нейронної мережі, щоб вона видавала кращий прогноз для конкретної задачі, що означає, що при правильному підборі активаційних функцій модель буде видавати менше помилок.

Функція “softmax” зазвичай використовується у вихідному шарі. Добре справляється з багато класовою класифікацією.

Функція “sigmoid” переводить вхідні значення з проміжку від мінус нескінченності до плюс нескінченності у значення з відрізка від нуля до

одного. Не рекомендується використовувати для регресії. Ця функція може призводити до проблеми зникаючих градієнтів, що призводить до зниження точності моделі, а також займає багато часу для її навчання.

Функція “tanh” є нелінійною, переводить вхідні значення у вихідні з діапазону від мінус одного до плюс одного. Вона також призводить до проблеми зникаючих градієнтів. Використовується зазвичай у так званих рекурентних нейронних мережах.

Варто зауважити, що останні дві функції не можна використовувати у моделі з великою кількістю шарів через вказану проблему.

Активаційна функція “relu” перетворює негативні вхідні значення нейронів у нуль, в той час як позитивні залишає незмінними. Вона проста, а значить, і швидка, у обчисленнях та не призводить до проблеми зникаючих градієнтів, на відміну від минулих функцій. Але при занадто швидкому навчанні, ваги можуть змінитися на значення, яке призведе до того, що нейрон не буде оновлюватися в жодній точці знову. Це означає, що нові ітерації не принесуть ніякої користі, оскільки результат залишиться незмінним. Дана функція використовується тільки для прихованих шарів, для вихідного ж використовують інші, наприклад, вказані раніше.

При підборі параметрів для нейронної мережі можна спробувати також різні модифікації функції “relu”, такі як “leaky relu”, “prelu” та “maxout” [8].

Отже, використовувати нейронні мережі для різних задач, пов’язаних з обробкою зображень, є ефективним і популярним способом вирішення поставленої проблеми. Згорткові нейронні мережі найкраще справляються з обробкою зображень, а їх поєднання з іншими моделями дозволяє просто та швидко отримати бажаний результат, наприклад, класифікувати зображення за певним критерієм.



## **РОЗДІЛ 2: ЗАСТОСУВАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ПОМИЛОК У ДОКУМЕНТАХ**

### **2.1 Постановка задачі**

Для відбору задачі, яку буде розв'язано у рамках даної курсової роботи, було проведено аналіз декількох варіантів реальних проблем, з якими стикаються різні компанії, зокрема компанія “IT-Enterprise”. Вони мають декілька реальних задач, які гарно було б автоматизувати, оскільки їх вирішення співробітниками займає багато часу та не звільняє від помилок, які важко контролювати. Прикладами таких задач є розпізнавання печаток (чи їхню відсутність) у документах, або розпізнавання підписів, або ж виявлення російських слів у україномовному документі.

Всі ці задачі можливо вирішити засобами машинного навчання.

В рамках виконання курсової роботи обрано задачу розпізнавання печаток у документах. Prozorro - сервіс проведення публічних закупівель, де державні організації-замовники оприлюднюють тендерні оголошення, а учасники аукціону беруть участь у торгах, при цьому, на відміну від звичайних аукціонів, тут виграє той учасник, який поставить найменшу ціну. Даний сервіс забезпечує здорову екосистему публічних закупівель, підтримуючи прозорість усього їхнього процесу, а також запобігає корумпованим угодам. Також серед його особливостей можна виділити те, що будь-які дані, розміщені на сайті, є дозволеними для їх подальшого використання та розповсюдження [9].

SmartTender використовує правила Prozorro для організації торгів, а також є однією з площадок компанії “IT-Enterprise”, при підтриманні якої вони

зіштовхнулися з проблемою розпізнавання, в яких документах присутні печатки, оскільки їх відсутність, за правилами користування даним ресурсом, є порушенням правил проведення закупівель, а отже, є важливим аспектом розпізнавання таких угод задля вжиття певних заходів.

Тут варто враховувати, що документи представлені не тільки в форматі зображень (з розширенням “.png”, “.jpg” або “.jpeg”), а також у вигляді документів PDF. Тому при формуванні навчальної вибірки (а також при подальшій класифікації документів, що мають печатку, або не мають її взагалі), слід перетворити усі документи, представлені у форматі PDF, на зображення.

Для навчання обраної моделі машинного навчання, на вхід подаються зображення документів, які вона класифікує як 0, якщо на зображенні немає печатки, та як 1, якщо має хоча б одну.

## **2.2 Існуючі методи**

При вивченні даної теми виявилось, що існують різні методи, які використовуються для розпізнавання печаток на документах. Деякі з них представлені далі.

Двоетапний підхід до вилучення візуальних об'єктів з паперових документів [10] серед інших задач вирішує і задачу розпізнавання печаток у документах. Двоетапний підхід працює наступним чином: спочатку застосовується певний алгоритм для розпізнавання об'єкту на вхідному зображенні, після чого застосовується інший метод, заснований на добуванні з зображення низько-рівневих характеристик (з англ. “features”) - таким чином перевіряється правильність роботи попереднього етапу. Отже, перший етап - каскадне навчання та розпізнавання на основі класифікатора AdaBoost.

На другому етапі проводиться оцінка низько-рівневих характеристик зображення за допомогою різних алгоритмів машинного навчання. Зображення представляється даними, отриманими з нього за допомогою різних функцій, однією з яких є статистика першого порядку, що означає представлення зображення у вигляді таких характеристик як середня інтенсивність пікселів, дисперсія, асиметрія, центральний момент та ентропія. Наступним методом представлення зображення є його опис за допомогою статистики довжини сірого, дані про яку надають інформацію щодо текстури зображення. Також серед таких методів є гістограма напрямлених градієнтів, яка допомагає розрізняти об'єкти різних типів, та локальні бінарні патерни, які є універсальними дескрипторами текстури. На другому етапі до усіх цих представлень застосовуються алгоритми машинного навчання, такі як метод  $k$ -найближчих сусідів ( $k=1$ ), наївний Байєс, метод опорних векторів, бінарне дерево рішень та інші. Використання двоетапного підходу до вилучення візуальних об'єктів з паперових документів у випадку розпізнавання печаток дало середню точність 53.3%.

Наступний підхід до виявлення печаток у документах використовує поєднання деяких простих характеристик зображення. Алгоритми машинного навчання (такі як метод  $k$ -найближчих сусідів, метод опорних векторів, випадкові ліси), що використовуються для виявлення печаток, обробляють інформацію про зображення, в якому закодували початкову модель RGB у модель, що представляє зображення як поєднання  $Y$ ,  $C_b$ ,  $C_r$ , де кожна з компонент є сумою значень RGB, перемножених на сталі коефіцієнти, після чого бінаризували зображення. В результаті навчання та валідації зображень, виявилось, що середня точність передбачень становить близько 70% [11].

Також існує підхід до розпізнавання печаток, який в цілому фокусується на розпізнаванні геометричних форм, притаманних їм. Для цього використовується перетворення Хафа, оскільки його метою є виявлення кругів та квадратів. Також тут застосовується алгоритм згладжування, який прибирає шуми. Після усіх перетворень метод опорних векторів класифікує документи за наявністю/відсутністю печаток. За словами авторів, вони досягли 92% точності роботи алгоритму [12]. Але тут варто зауважити, що даний підхід фокусується на розпізнаванні печаток, представлених у формі круга чи квадрату, що є лише підмножиною усіх можливих форм печаток. А тому результати дослідження є ідеалізованими і не співпадають реальним практичним задачам.

## **2.3 Імплементация методу**

Розв'язувати задачу розпізнавання печаток у документах було вирішено за допомогою згорткових нейронних мереж, які є найефективнішим методом роботи з зображеннями. Але перед використанням нейронних мереж для класифікації документів необхідно сформувати тренувальну та тестову вибірки.

### **2.3.1 Завантаження даних**

Оскільки дані для роботи відсутні, необхідно завантажити певну кількість документів для подальшої обробки та класифікації.

Prozorro має відкриті дані по закупівлям, в тому числі і документи, які вільні для використання завдяки політиці компанії. Для початку був проведений аналіз наявних документів. Виявлено, що усі дані, які підходять для вирішення даної задачі, представлені або як зображення, або як документи у форматі PDF. Документи по закупівлям, що мають текстовий



формат (тобто, розширення “.doc” або “.docx”), не підходять, оскільки зазвичай є просто деталями договору, підготовленими на друк, та, відповідно, не містять печаток взагалі.

Для вилучення даних в автоматизованому режимі, за допомогою засобів мови програмування Python для веб-скрапінгу, необхідно проаналізувати посилання на сторінки, з яких будуть завантажуватися документи.

В ході такого аналізу було виявлено, що посилання представлені у форматі

“https://prozorro.gov.ua/tender/UA-{year}-{mon}-{day}-{uid}-{s}/”,

де year - це рік проведення закупівель, mon - місяць, day - день, uid - ідентифікатор події, s - літера зі списку (“a”, “b”, “c”), причому до місяця і дня, якщо вони належать проміжку [1, 9], дописується нуль у початок значення, а uid є шестизначним і також містить нулі у початку, якщо значення є п’ятизначним і менше.

Оскільки на сторінці, що містить список тендерів, представлено всього десять перших, а для розширення списку використовується кнопка “Показати більше результатів”, з обробкою якої виникли проблеми, було вирішено шукати перебором можливих посилань ті, що мають статус “Завершена”, для отримання посилань на документи. Реалізація даної функції представлена у Додатку 1.

Після отримання файлу зі списком посилань на скачування документів, необхідно їх завантажити та видалити не потрібні.

```

def docs_download(links):
    with open(links, "rb") as f:
        ls = pickle.load(f)

    folder = f'{os.path.splitext(links)[0]}'
    Path(folder).mkdir(parents=True, exist_ok=True)

    for i, l in enumerate(ls):
        url = l

        remotefile = urlopen(url)
        info = remotefile.info()['Content-Disposition']
        value, params = cgi.parse_header(info)

        ext = os.path.splitext(params["filename"])[-1]

        urlretrieve(url, f'{folder}/{i}{ext}')

```

Рис. 2.1 - Функція завантаження документів за посиланнями

В функції docs\_download() зчитується файл, який містить список усіх посилань, після цього за назвою файлу створюється директорія і у циклі наповнюється завантаженими документами з оригінальним розширенням та назвою файлу, що відповідає поточній ітерації.

```

def docs_filter(folder_name):
    all_files = os.listdir(folder_name)
    filtered = [file for file in all_files if file.endswith(".p7s")
               or file.endswith(".doc") or file.endswith(".docx")]

    for file in filtered:
        path = os.path.join(folder_name, file)
        os.remove(path)

```

Рис. 2.2 - Функція фільтрування документів

Функція docs\_filter() зчитує назви усіх документів, завантажених у певну директорію, утворює список з усіма файлами, які мають розширення “.p7s”, “.doc” або “.docx”, та видаляє їх. В результаті виконання цієї функції в

директорії залишаються тільки необхідні для розв'язання поставленої задачі документи.

### 2.3.2 Попередня обробка документів

Для забезпечення правильної роботи класифікатора необхідно всі документи, представлені як документи PDF, перетворити на зображення. Для цього створено функцію `pdf2image()`.

```
def pdf2image(filenamees_list, folder_name):  
    for file in filenamees_list:  
        images = convert_from_path(f'{folder_name}/{file}')  
        filename = os.path.splitext(file)[0]  
        #print(filename)  
  
        for i, image in enumerate(images):  
            im = image.save(f'{folder_name}/{filename}_{i}.png')
```

Рис. 2.3 - Функція конвертації файлів з вхідного списку у зображення

Вона обробляє список з назвами файлів, які необхідно конвертувати, у циклі перетворює їх на зображення, та ці зображення, кожне з яких відповідає певній сторінці конвертованого документу, у іншому циклі зберігаються у поточну папку.

Для перетворення усіх PDF-документів представлена вище функція викликається наступною.

```
def all_pdf2image(folder_name):  
    all_files = os.listdir(folder_name)  
    filtered = [file for file in all_files if file.endswith(".pdf")  
                or file.endswith(".PDF")]  
  
    pdf2image(filtered, folder_name)
```

Рис. 2.4 - Функція перетворення усіх PDF-документів даної директорії у зображення

Тут для вказаної директорії з усіх файлів обираються тільки ті, що мають необхідне розширення файлу, та до них застосовується попередня функція, яка конвертує усі файли зі списку в зображення.

Після конвертації документів у зображення, необхідно відокремити файли, які будуть використовуватися надалі.

```
def form_images_folder(folder_name):
    all_files = os.listdir(folder_name)
    images = [file for file in all_files if file.endswith(".png")
              or file.endswith(".jpg") or file.endswith(".jpeg")]

    #return all formats
    #result: ['.pdf', '.PDF', '.png', '.jpg']
    #print(list(set([os.path.splitext(file)[-1] for file in all_files])))

    Path(f'{folder_name}/images').mkdir(parents=True, exist_ok=True)

    for i in images:
        shutil.move(f'{folder_name}/{i}', f'{folder_name}/images/{i}')
```

Рис. 2.5 - Функція формування директорії з усіма зображеннями

Дана директорія з зображеннями формується для подальшого розбиття на навчальну та тестову вибірки. Але перед цим необхідно стандартизувати розмір зображень, оскільки для застосування алгоритмів машинного навчання на них, зображення мають бути однакової роздільної здатності.

```
def resize(folder_name):
    images = os.listdir(folder_name)

    resized_folder = 'links_01_04_2020_uid100/resized'
    Path(resized_folder).mkdir(parents=True, exist_ok=True)

    for i in images:
        image = cv2.imread(f'{folder_name}/{i}')
        resized = cv2.resize(image, (1350, 1900), interpolation = cv2.INTER_AREA)
        image = cv2.imwrite(f'{resized_folder}/{i}', resized)
```

Рис. 2.6 - Функція перетворення зображень на ті, що мають роздільну  
здатність 1350x1900

Результатом виконання даної функції є утворення директорії, яка містить усі зображення зі стандартним розміром.

Тепер відбувається процес розбиття вхідних даних на тренувальну та тестову вибірки. Для цього визначається розмір тренувальної вибірки та створюється список, випадковим чином наповнений індексами файлів з директорії. Після чого створюються списки з індексами зображень, які увійшли до тренувальної та тестової вибірок.

```

random_range = len(images)
train_size = int(0.7 * random_range) #set number of elements for training model

random_list = random.sample(range(0, len(images)), len(images))

train_list = random_list[0:train_size]
valid_list = random_list[train_size:]

```

Рис. 2.7 - Формування списків з індексами зображень для тренувальної та  
тестової вибірок

Наступним кроком формуються директорії для вибірок, що обговорюються вище.

```

#create 'train' folder
train_folder = '../stamps_detection/train'
if Path(train_folder).exists() and Path(train_folder).is_dir():
    pass
else:
    Path(train_folder).mkdir(parents=True, exist_ok=True)

#create 'validation' folder
valid_folder = '../stamps_detection/validation'
if Path(valid_folder).exists() and Path(valid_folder).is_dir():
    pass
else:
    Path(valid_folder).mkdir(parents=True, exist_ok=True)

```

Рис. 2.8 - Створення директорій для тренувальної та тестової вибірок

І усі зображення розбиваються по двом директоріям, при цьому додаючи назву файлу у відповідний список.

```

tname_list = []
vname_list = []

for ti in train_list:
    shutil.copy(f'{folder_name}/{images[ti]}', f'{train_folder}/{images[ti]}')
    tname_list.append(images[ti])
for vi in valid_list:
    shutil.copy(f'{folder_name}/{images[vi]}', f'{valid_folder}/{images[vi]}')
    vname_list.append(images[vi])

```

Рис. 2.9 - Наповнення тренувальної та тестової вибірок зображеннями

Для можливості використання машинного навчання при вирішенні поставленої проблеми, необхідно створити таблицю відповідності зображень класам.

```

#fill 'target' column with 0s but fill target values later by hands
tname_df = pd.DataFrame({'filename': tname_list,
                        'target': [0] * len(tname_list)},
                        columns=['filename', 'target'])
vname_df = pd.DataFrame({'filename': vname_list,
                        'target': [0] * len(vname_list)},
                        columns=['filename', 'target'])
tname_df.to_csv(f'{train_folder}/train.csv')
vname_df.to_csv(f'{valid_folder}/validation.csv')

```

Рис. 2.10 - Створення таблиць відповідності класів зображенням

В результаті виконання даного шматка коду створюються таблиці з назвами зображень, та відповідним їм класам. В даному випадку, всі цільові значення прирівнюються нулю, оскільки роботу над класифікацією зображень ще належить провести.

Приклади сторінок документів з печатками наведені у Додатках 2-5, без печаток - у Додатках 6-8: їх було вивантажено з сайту Prozorro.

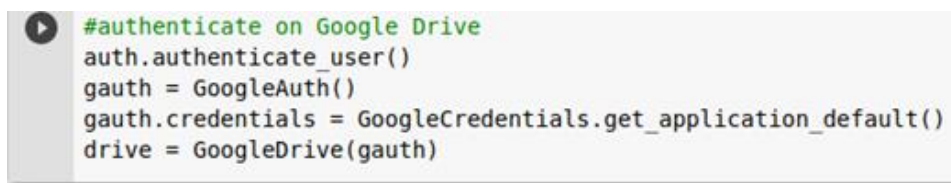


Після виконання усіх кроків попередньої обробки документів, можна переходити до використання згорткових нейронних мереж та класифікації зображень.

### 2.3.3 Класифікація документів за наявністю печатки

Навчання нейронної мережі на зібраних даних було проведено на безкоштовному хмарному сервісі Google Colab, створеного на основі Jupyter Notebook.

Для роботи з даними їх було завантажено на Google Drive у вигляді архівів, після чого за допомогою ідентифікатора до них можна доступатися та завантажувати у віртуальний файловий простір. Для цього треба пройти аутентифікацію за допомогою акаунту Google.



```
#authenticate on Google Drive
auth.authenticate_user()
gauth = GoogleAuth()
gauth.credentials = GoogleCredentials.get_application_default()
drive = GoogleDrive(gauth)
```

Рис. 2.11 - Аутентифікація користувача

Після виконання попереднього кроку треба завантажити файли у віртуальний файловий простір та розархівувати їх.



```
# Get train and validation data from GD
train = drive.CreateFile({'id': '1ESU6W-8ugAsBPPC0QRANHEbEp0CyQIVT'})
train.GetContentFile('train.zip')

valid = drive.CreateFile({'id': '130pwjSebfaHkTA-3ITr-NjWbbiyxpKE8'})
valid.GetContentFile('validation.zip')

!unzip train.zip
!unzip validation.zip
```

Рис. 2.12 - Завантаження даних

Коли дані завантажені, таблиця з тренувальними даними оброблена та розбита на тренувальну та валідаційну вибірки, будується нейронна мережа та проводиться її навчання.

```
# build convolutional neural network and train it
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.layers import Conv2D, MaxPooling2D

model = Sequential()
model.add(Conv2D(64, kernel_size=3, activation='relu', input_shape=(28,32,3)))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(1000, activation='relu'))
model.add(Dropout(0.25))
model.add(Dense(500, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy',
              optimizer='Adam', metrics=['accuracy'])

model.fit(X_train, y_train, epochs=20, validation_data=(X_test, y_test))
```

Рис. 2.13 - Реалізація нейронної мережі для класифікації документів за наявністю або відсутністю печаток

Для класифікації документів використовується послідовна модель, яка у своїй структурі має три згорткових шарів для обробки зображень, та використовує ще три для їхньої класифікації.

До усіх шарів, окрім вихідного, застосовується активаційна функція “relu”, останній же використовує “sigmoid”. При заміні активаційної функції у вихідному шарі (зокрема, на активаційну функцію “softmax”) точність роботи моделі зменшується приблизно у два рази.

## 2.4 Результати



Отже, перша спроба класифікувати документи на тестовому наборі дала точність приблизно у 81%. При цьому для навчання моделі використовувалося 544 зображення (при навчанні нейронної мережі 435 використовувалися саме для навчання та 109 для валідації роботи моделі). Тестовий набір містив 234 зображень документів.

Щоб збільшити точність роботи моделі, було збільшено тренувальну вибірку.

### **2.4.1 Залежність точності моделі від кількості навчальної вибірки**

Для порівняння правильності роботи моделі в залежності від кількості тренувальних даних, тестування проводилося на одному й тому самому наборі зображень. Це дає можливість побачити вплив більшої кількості даних на більш правильне налаштування моделі під час навчання.

Після отримання даних для навчання, набір поділяється на тренувальну та валідаційну вибірки, які використовуються для навчання моделі. Набір даних для валідації роботи моделі складає близько 25% від загального об'єму тренувальних даних.

Під час навчання моделі два параметри мають важливе значення для характеристики нейронної мережі. Це “loss” (від англ. “втрати”) та “accuracy” (від англ. “точність”). Чим нижче значення першого параметра і більше - другого, тим краще працює модель. До того ж близькі значення цих параметрів для тренувальної та валідаційної вибірок означають гарно-навчену модель, без “over-fitting” та “under-fitting” (вказують на перенавчання моделі та, навпаки, на погано навчену модель відповідно).

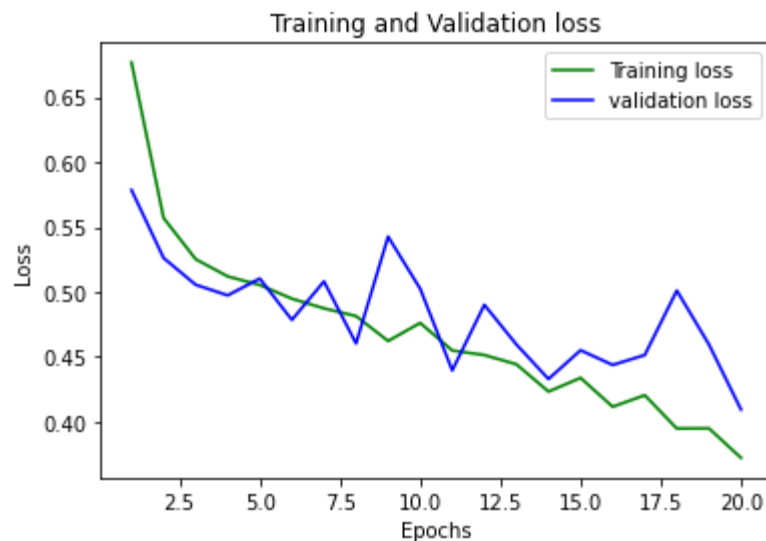


Рис. 2.14 - Втрати в процесі тренування та валідації моделі

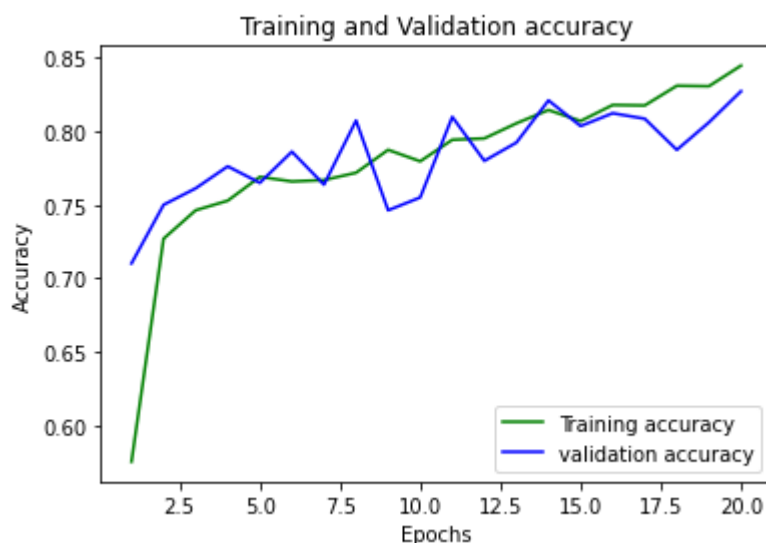


Рис. 2.15 - Точність в процесі тренування та валідації моделі

Під час тестування роботи моделі, яку було навчено на 435 зображеннях, виявилося, що правильно класифіковано приблизно 81% тестових даних. А після навчання нейронної мережі на 3216 об'єктах, доля правильних відповідей становила трохи більше 88%.

Також варто зауважити, що на другій і третій ітерації точність роботи моделі на тестових даних трохи погіршилась, після чого досить різко виросла на двох останніх. Такий ефект можна приписати якості самих даних та їхньої попередньої класифікації, оскільки вона проводилася вручну, а при цьому можлива невелика похибка. Щодо самих даних, то при класифікації було виявлено досить велику кількість погано відсканованих документів, на яких відображалися печатки з наступних сторінок. В результаті було вирішено класифікувати такі зображення як ті, що не містять печаток, хоча їх на документі гарно видно.

<b>Розмір тренувальної ї вибірки</b>	<b>Розмір валідаційної ї вибірки</b>	<b>Розмір тестової ї вибірки</b>	<b>Точність роботи моделі</b>
<b>435</b>	109		<b>81.19%</b>
<b>1164</b>	291		<b>80.76%</b>
<b>1896</b>	474	234	<b>80.76%</b>
<b>2596</b>	650		<b>86.32%</b>
<b>3216</b>	804		<b>88.03%</b>

Рис. 2.16 - Вплив розміру тренувального набору даних на точність роботи нейронної мережі

При збільшенні розміру тренувальної вибірки час на навчання моделі зростає, а час роботи на тестових даних починає істотно збільшуватися тільки на останніх ітераціях.

Розмір тренувальної вибірки	Розмір валідаційної вибірки	Розмір тестової вибірки	Час, необхідний на навчання моделі (у хв)	Час роботи мережі на тестових даних (у сек)
435	109		0,66	0,23
1164	291		1,78	0,22
1896	474	234	2,86	0,21
2596	650		3,86	0,27
3216	804		4,81	0,37

Рис. 2.17 - Вплив розміру тренувального набору даних на час роботи  
нейронної мережі

## Висновки

Машинне навчання широко використовується для роботи з зображеннями, особливо ефективним підходом до їхньої обробки є використання згорткових нейронних мереж. Через структуру архітектури мереж даного типу вони гарно працюють з зображеннями, тому широко використовуються для роботи з цим типом даних.

Автоматизоване розпізнавання різних типів помилок у документах є актуальною темою, що потребує вирішення. Прикладом таких помилок є відсутність печаток на документах, для розпізнавання чого в рамках даної курсової роботи було побудовано нейронну мережу, здатну класифікувати дані, що подаються на вхід у вигляді зображень, за наявністю або відсутністю печаток.

Початкова точність роботи даної моделі складала 81.19%, при цьому для її навчання використовувалося 435 зображень (для тренування мережі) і 109 (для валідації її роботи). Наступні дві спроби навчити модель на більшій кількості даних, ніж при попередній спробі, дали трохи гірші результати, а саме точність роботи моделі на тестовому наборі даних складала 80.76% в обох випадках. Спочатку для навчання мережі використовувалося 1164 зображення (і 291 для валідації), після чого на вхід моделі було подано 1896 зображень для навчання (і 474 для валідації). Останні два процеси навчання на більшій кількості даних (2596 для навчання і 650 для валідації в першому випадку, 3216 для навчання і 804 для валідації - в другому), показали покращення при класифікації тестового набору, що дає 86.32% і 88.03% правильних відповідей.

Отже, в результаті виконання даної курсової роботи були отримані такі наукові результати:

1. Вивчено різні методи роботи з зображеннями.
2. Досліджено можливість використання машинного навчання для роботи з зображеннями.
3. Проведено аналіз існуючих методів вирішення поставленої задачі.
4. Проведено збір та обробку даних.
5. Побудовано нейронну мережу для класифікації документів.
6. Вивчено залежність точності класифікації від кількості даних для навчання моделі.

## Список використаних джерел

1. Tyagi V. Understanding Digital Image Processing / Vipin Tyagi., 2018.
2. Azad M. Color Image Processing in Digital Image / M. Azad, M. Hasan, M. Naseer. // International Journal of New Technology and Research (IJNTR). – 2017. – №3. – С. 56–62.
3. Otsu N. A threshold selection method from gray-level histograms / N. Otsu. // IEEE Trans. Systems, Man, and Cybernetics. – 1979. – №9(1). – С. 62–66.
4. Ishabailu F. Image Noise Reduction and Filtering Techniques / F. Ishabailu, A. Hambal, Z. Pei. // International Journal of Science and Research (IJSR). – 2017. – №3.
5. What Is Image Segmentation? 3 things you need to know [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ch.mathworks.com/discovery/image-segmentation.html>.
6. Saha S. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way [Електронний ресурс] / Sumit Saha. – 2018. – Режим доступу до ресурсу: <https://ch.mathworks.com/discovery/image-segmentation.html>.
7. Neural Network Diagram [Електронний ресурс] – Режим доступу до ресурсу:  
[https://www.astroml.org/book\\_figures/chapter9/fig\\_neural\\_network.html](https://www.astroml.org/book_figures/chapter9/fig_neural_network.html).
8. Jain P. Complete Guide of Activation Functions [Електронний ресурс] / Pawan Jain // 2019 – Режим доступу до ресурсу:  
<https://towardsdatascience.com/complete-guide-of-activation-functions-34076e95d044>.
9. Офіційний портал оприлюднення інформації про публічні закупівлі України [Електронний ресурс] – Режим доступу до ресурсу:  
<https://prozorro.gov.ua/about>.

10. Forczmanski P. Two-stage approach to extracting visual objects from paperdocuments / P. Forczmanski, A. Markiewicz. // Machine Vision and Applications. – 2016. – №27. – С. 1243–1257.
11. Forczmanski P. Stamps Detection and Classification Using Simple Features Ensemble / P. Forczmanski, A. Markiewicz. // Mathematical Problems in Engineering. – 2015.
12. Roy P. Seal Detection and Recognition: An Approach for Document Indexing [Электронный ресурс] / P. Roy, U. Pal, J. Lladós // 10th International Conference on Document Analysis and Recognition. – 2009. – Режим доступа до ресурсу:  
[https://www.researchgate.net/publication/220861099\\_Seal\\_Detection\\_and\\_Recognition\\_An\\_Approach\\_for\\_Document\\_Indexing](https://www.researchgate.net/publication/220861099_Seal_Detection_and_Recognition_An_Approach_for_Document_Indexing).
13. original image: Lena [Электронный ресурс] – Режим доступа до ресурсу:  
<https://www.cosy.sbg.ac.at/~pmeerw/Watermarking/lena.html>.



## Додаток 1

```
def get_links(filename):
```

```
    url = 'https://prozorro.gov.ua/tender/'
```

```
    global_links = []
```

```
    for m in range(4, 5):
```

```
        for d in range(1, 2):
```

```
            for i in range(1, 100):
```

```
                for s in "abc":
```

```
                    day = '{:02}'.format(d)
```

```
                    mon = '{:02}'.format(m)
```

```
                    uid = '{:06}'.format(i)
```

```
                    new_url = f'{url}UA-2020-{mon}-{day}-{uid}-{s}/'
```

```
                    session = dryscrape.Session()
```

```
                    session.visit(new_url)
```

```
                    time.sleep(5)
```

```
                response = session.body()
```

```
                soup = BeautifulSoup(response)
```

```
            try:
```

```
                status = soup.find('span', class_='marked').get_text()
```

```
            except AttributeError:
```

```
                status = "Non"
```

```
    if status == 'Завершена':
```

```

        tables = soup.findAll('table', class_='table table-striped margin-
bottom prev')
    for t in tables:
        if len(t.findAll('a')) > 1:
            links = t.findAll('a')
            links.pop(0)
        elif len(t.findAll('a')) == 1:
            links = t.findAll('a')

    for l in links:
        global_links.append(l['href'])

with open(filename, "wb") as f:
    pickle.dump(global_links, f)

```

## Додаток 2

Додаток №1  
до Договору постачання № 230  
від «17» 04 2020 р.

м. Харків

«    »    2020р.

## Специфікація

№ з/п	Найменування	Од. виміру	Кіл-сть	Ціна за од., грн., без ПДВ	Сума, грн., без ПДВ
1	3М 5911 Фільтр від пилу/аерозолів Р1 (компл.2од.)	Компл.	120,00	27,40	3288,00
2	Фільтр вугільний від органічних парів DR 0047 аналог 3м 6051-A1 (компл.2од.)	Компл.	30,00	224,90	6747,00
				Всього:	10035,00
				ПДВ 20%:	2007,00
				Разом:	12042,00

Загальна вартість Товару становить: 10035,00грн (десять тисяч тридцять п'ять грн., крім того ПДВ – 2007,00 грн., разом з ПДВ – 12042,00 грн. (дванадцять тисяч сорок дві грн. 00 коп.).

Дана специфікація складена в двох примірниках, які мають однакову юридичну силу, і є невід'ємною частиною договору.

## ПОСТАЧАЛЬНИК:

ТОВ «Востоквнешторг Стальсервіс»

61001, місто Харків, майдан Повстання будинок 7/8

Тел. (057) 728-30-54

п/р UA 423808050000000026003548705 в АТ

«Райффайзен Банк АВАЛЬ», м. Харків.

МФО 380805,

код за ЄДРПОУ 34390588,

ІПН 343905820383, св. ПДВ №100262387

Тел. (057) 728-30-54

## ПОКУПЕЦЬ:

Державне підприємство «Харківський завод спеціальних машин»

Юридична адреса: 61045, м. Харків, вул. Дербентська, 109

Почтова адреса: 61045 м. Харків, вул. Дербентська, 109

UA513518230000026007300977468 в філії ХОУ АТ

«Ошадбанк», м. Харків

ЄДРПОУ 07904041, ІПН 079040420398, свідоцтво №200035645,

МФО351823

Тел. приймальня (057) 314-00-09,

тел. бухгалтерія (057) 341-00-55,

відділ постачання (057) 341-00-87, 341-00-37.



## Додаток 3

Додаток № 1  
до Договору поставки  
№ 191-Т/2020 від «01» 04 2020 року

## Специфікація

Код ДК 021:2015: 44110000-4 – конструкційні матеріали

№	Асортимент (номенклатура) товару	Одиниця виміру	Кіль- кість	Ціна за одиницю без ПДВ, грн.	Загальна вартість товару
1	2	3	4	5	6
1	Ферозіт 100/25 кг клей для плитки	шт	6	99.00	594.00
2	Клини до плитки 100 шт	шт	1	12.00	12.00
3	Хрестики 2мм	шт	2	13.00	26.00
4	Маркер міні STANLEY	шт	1	19.00	19.00
5	Цемент 500/25 кг	шт	5	95.00	475.00
6	Рейка шпукатурна маяк 6 мм 3 метра	шт	4	11.00	44.00
7	Плитка Санвуд 18,5*59,8 Вайт Церсаніт	шт	23	253.00	5 819.00
				<b>ВСЬОГО:</b>	<b>6 989.00</b>

## ПОКУПЕЦЬ:


Комунальне некомерційне підприємство  
«Нововолинський Центр первинної медико  
санітарної допомоги Нововолинської міської  
ради Волинської області»

Головний лікар  О.О. Попіка

«01» 04 2020 року  
М.П.

## ПРОДАВЕЦЬ:

Фізична особа-підприємець  
Селедець Олег Володимирович

 О.В. Селедець  
«01» 04 2020 року  
М.П.

## Додаток 4

Договору:

6.3.3. Відвантаження та доставку продукції проводити згідно зі специфікацією власним транспортом, або транспортом фірм-перевізників за власний рахунок.

**6.4. Постачальник має право:**

6.4.1. Своєчасно та в повному обсязі отримувати плату за поставлений товар;

6.4.2. На дострокову поставку товарів за письмовим погодженням Замовника;

6.4.3. У разі невиконання зобов'язань Замовником Постачальник має право достроково розірвати цей договір, повідомивши про це Замовника у строк трьох робочих днів.

**7. Відповідальність сторін**

7.1. У разі невиконання або неналежного виконання своїх зобов'язань за Договором Сторони несуть відповідальність, передбачену законами та цим Договором.

7.2. У разі порушення умов зобов'язання щодо якості (комплектності) товарів (робіт, послуг) Постачальником сплачується Замовнику штраф у розмірі 20% вартості неякісних (некомплектних) товарів (робіт, послуг).

7.3. У разі порушення строків виконання зобов'язання Постачальником сплачується Замовнику пеня у розмірі 0,1% вартості товарів (робіт, послуг) за кожний день прострочення, а за прострочення понад тридцять днів додатково сплачується штраф у розмірі семи відсотків вказаної вартості.

7.4. У разі порушення строків виконання зобов'язання при закупівлі товарів за бюджетні кошти Замовником, Постачальник має право стягнути з Замовника 0,1% від суми яка належить до сплати Постачальнику, у разі якщо Замовник не доведе поважність причини не своєчасної сплати коштів при наявності відповідного фінансування.

**8. Обставини непереборної сили**

8.1. Сторони звільняються від відповідальності за невиконання або неналежне виконання зобов'язань за цим Договором у разі виникнення обставин непереборної сили, які не існували під час укладання Договору та виникли поза волею сторін (аварія, катастрофа, стихійне лихо, епідемія, епізоотія, війна тощо).

8.2. Сторона, що не може виконувати зобов'язання за цим Договором унаслідок дії обставин непереборної сили, повинна не пізніше ніж протягом трьох робочих днів з моменту їх виникнення повідомити про це іншу Сторону у письмовій формі.

8.3. Доказом виникнення обставин непереборної сили та строку їх дії є відповідні документи, які видаються уповноваженими органами.

8.4. У разі коли строк дії обставин непереборної сили продовжується більше ніж тридцять днів, кожна із Сторін у встановленому порядку має право розірвати цей Договір.

**9. Вирішення спорів**

9.1. У випадку виникнення спорів або розбіжностей Сторони зобов'язуються вирішувати їх шляхом взаємних переговорів та консультацій.

9.2. У разі недосягання Сторонами згоди спори (розбіжності) вирішуються у судовому порядку.

**10. Строк дії договору**

10.1. Цей договір набирає чинності з дати підписання і діє до 01 травня 2020 року.

10.2. Цей договір складено у трьох примірниках, що мають однакову юридичну силу.

**11. Інші умови**

немає

**12. Додатки до договору**

Невід'ємною частиною цього Договору є специфікація договору.

**13. Місцезнаходження та банківські реквізити сторін**

**Замовник**

Військова частина 2196  
(Подільський прикордонний загін)  
66300 Одеська обл., м. Подільськ,  
вул. Армійська, 21  
IBAN UA 248201720343160001000006075  
МФО 820172, код ЄДРПОУ 14321825  
Державна казначейська служба України  
КИКВ 1002030/01-КЕКВ 2230

Командир військової частини 2196

Андрій САВЧУК

**Постачальник**

ФОП «Албанський О.Б.»  
66300, Одеська обл., м. Подільськ,  
вул. Соборна, 2166, ЄДРПОУ 2647323921  
IBAN UA 493510050000026004032445100

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

АТ «УКРАЇНСЬКА

Олег АЛБАНСЬКИЙ

## Додаток 5

Додаток № 3  
до Договору № 20 (С) від  
«01» квітня 2020 р.

**ЗАТВЕРДЖЕНО:**  
М.Я. Семко М.Я.  
(підпис) (прізвище, ініціали, посада)  
«01» квітня 2020 р.

**КОШТОРИС № 1**  
**на послуги, які надаються за Договором**  
**№20 (С) від «01» квітня 2020 р.**

Лісництво Остерське

квартал 85 виділ 2,1 площа 2,5 га

лісорубний квиток № 005328 від 23.03.2020р.

ліквідний запас 395кбм., в т.ч. ділової 324кбм., дров 71кбм.

\*(розробляється «Виконавцем» на визначений «Замовником» перелік послуг)

№ п/п	Шифр	Найменування послуг*	Один. виміру	Кількість	Розцінка за одиницю, грн.	Сума, грн.
А	Б	В	Г	Д	Е	Є
1.		Підготовчі роботи та допоміжні роботи на лісосіці + очистка лісосіки	м <sup>3</sup>	395	29,33	11586,67
1.1						
2.		Лісосічні роботи				
2.1		Комплексна заготівля:	м <sup>3</sup>	395	54,61	21569,0
		- круглих лісоматеріалів	м <sup>3</sup>	324	58,64	18999,0
2.2		- дров'яної деревини	м <sup>3</sup>	71	36,20	2570,0
3.		Трелювання деревини	м <sup>3</sup>	395	41,77	16501,0
3.1						
		<b>Всього вартість послуг</b>	м <sup>3</sup>	395	125,71	49656,67
		<b>ПДВ 20%</b>				9931,33
		<b>Вартість послуг з ПДВ</b>	м <sup>3</sup>	395	150,86	59588,0

Вартість послуг згідно договору становить п'ятдесят дев'ять тисяч п'ятсот вісімдесят вісім грн. 00 коп. В тому числі ПДВ дев'ять тисяч дев'ятсот тридцять одна грн. 33 коп.

(сума прописом)

«Виконавцем» визначено вартість знеособленого 1м<sup>3</sup> заготовленої (і зтрельованої) у розмірі 150,86 грн.

Кошторис розрахований відповідно до загальної вартості заготівлі (з ПДВ) : ЛМК- 162 грн. ; дрова  
деревина- 100 грн. додатково за відбір цінних сортментів (категорії А) – 10грн

Ознайомлений:

«Замовник» (підпис) Красна О.Л. гол. економіст

(підпис) (прізвище, ініціали, посада)

МП

«01» квітня 2020 р.



## Додаток 6

ЗВІТ  
про укладений договір

1. Дата укладення договору: 14.01.2020р..
2. Номер договору: №02/11-18 від 14.01.2020р..
3. Найменування замовника: ДП «АНТОНОВ».
4. Код згідно з ЄДРПОУ замовника: 14307529.
5. Місцезнаходження замовника: 03062, Україна, м. Київ, вул. Туполева, 1.
6. Найменування постачальника товарів, виконавця робіт чи надавача послуг (для юридичної особи) або прізвище, ім'я, по батькові (для фізичної особи), з яким укладено договір: Aerogulf FZC.
7. Код згідно з ЄДРПОУ/реєстраційний номер облікової картки платника податків постачальника товарів, виконавця робіт чи надавача послуг: відсутні.
8. Місцезнаходження постачальника товарів, виконавця робіт чи надавача послуг (для юридичної особи) або місце проживання (для фізичної особи) та номер телефону, телефаксу: P.O. Box 9204, Sharjah U.A.E., P6-157, SAIF Zone, SHARJAH, U.A.E. (м. Шарджа, Об'єднані Арабські Емірати).
- 8.1. Вид предмета закупівлі: роботи.
9. Конкретна назва предмета закупівлі: Роботи: «Послуги з експлуатації ДГТД ТА18-200-124 в складі літаків Ан-124-100».
10. Найменування (номенклатура, асортимент) товарів, робіт чи послуг: Роботи: «Послуги з експлуатації ДГТД ТА18-200-124 в складі літаків Ан-124-100».
11. Кількість товарів, робіт чи послуг: 1 робота.
12. Місце поставки товарів, виконання робіт чи надання послуг: аеропорт Лейпциг.
13. Строк поставки товарів, виконання робіт чи надання послуг: 30 календарних днів.
14. Інформація про технічні та якісні характеристики товарів, робіт чи послуг: у відповідності з технічними умовами на дані роботи.
15. Ціна договору: 41 177,00 дол. США (ПДВ не обкладається).
16. Строк дії договору: 31.12.2020р..
- 16.1. Джерело фінансування закупівлі: кошти підприємства.
17. Ідентифікатор договору\*.
18. Одиниця виміру - роботи.
19. Ціна за одиницю: 41 177,00 дол. США (ПДВ не обкладається)..

\* Не заповнюється (заповнюється електронною системою закупівель автоматично).

Начальник відділу організації  
ремонту комплектуючих

11.03.2020 10.03.20р. Є.Р. Діський  
підпис, дата ПІБ  
1.Р. Власенко

## Додаток 7

## 4. Доплати і нарахування

1. Компенсація витрат на відрядження експертів на засідання ДКЗ (без ПДВ):	0
2. Разом витрат:	312276,89
3. ПДВ (20%):	62455,38
4. ВСЬОГО:	374732,27
5. Надбавка за виконання робіт, не передбачених планом (%):	
6. Всього по договору:	374732 грн. 27 коп. (триста сімдесят чотири тисячі сімсот тридцять дві грн. 27 коп.), у т.ч. ПДВ – 62455 грн. 38 коп.

Начальник управління

Начальник відділу

Головний економіст

(О.В. Нещкий)

(О.О. Бакаржівса)

(Н. М. Лисенко)

*Примітка: \** Коефіцієнт повних затрат враховує витрати, пов'язані з організаційно-господарською діяльністю ДКЗ. Величина коефіцієнту повних витрат визначається як відношення загальних витрат ДКЗ до вартості експертизи. (Порядок визначення вартості робіт по проведенню державної експертизи та оцінки запасів корисних копалин, затверджений наказом ДКЗ України від 02.08.1996р. №29 та зареєстрований в Міністерстві юстиції України 05.11.1996 за № 652/1677).

Загальні витрати складаються з фонду оплати праці працівників ДКЗ, податків і обов'язкових платежів, витрат на комунальні послуги, зв'язок та інші послуги, використання та обслуговування техніки, організаційне та інформаційне забезпечення експертних робіт і визначаються як правило один раз на рік за фактичними економічними даними минулого року і тими, що плануються у поточному році. Згідно з виконаними підрахунками коефіцієнт повних витрат на 2019 р. дорівнює 7,7 (додаток до наказу ДКЗ України від 28.02.2019 р. № 108).



## Додаток 8

84544, 00

Договір 15-1020  
про закупівлю товарів за державні кошти

м. Одеса «01» 04 2020 р.

18 військова майстерня, в особі начальника Лекарева Віталія Ігоровича, який діє на підставі Положення про військове (корабельне) господарство Збройних Сил України, затвердженого наказом Міністра оборони України від 16.07.1997 №300 (далі - **Замовник**), з однієї сторони, і  
фізична особа-підприємець **Спіцин Олексій Володимирович** (РНОКПП - 2803813439), який діє на підставі Виписки з Єдиного державного реєстру юридичних осіб, фізичних осіб-підприємців та громадських формувань запис від 08.07.2015 № 2 556 000 0000 116860, та є платником єдиного податку, не платник ПДВ, (надалі - **«Продавець»**), з іншої сторони,  
разом - **Сторони**, уклали даний договір про наступне (далі - **Договір**):

**1. ПРЕДМЕТ ДОГОВОРУ**

1.1. Постачальник зобов'язується у 2020 році поставити Замовнику Товар, зазначений в п. 1.2. цього Договору, а Замовник - прийняти і оплатити такий Товар.

1.2. Найменування товару:  
спеціальний одяг робочий (код за ДК 021:2015 - 18130000-9, «Спеціальний робочий одяг»), на суму 55 110,00 грн.,  
спеціальне взуття захисне (код за ДК 021:2015 - 18830000-6, «Захисне взуття»), на суму 21 484,00 грн.,  
краги зварювальника, рукавиці брезентові (код за ДК 021:2015 - 18140000-2, «Акcesуари до робочого одягу»), на суму 7 950,00 грн.,  
(далі за текстом - **Товар**).

1.3. Кількість та асортимент Товару - відповідно до Специфікації Товару (додаток №1 до цього Договору) та Таблиці розмірів спеціального робочого одягу та спеціального взуття захисного (додаток №3 до цього Договору), які додаються до цього Договору і є його невід'ємними частинами.

1.4. Обсяги закупівлі Товару можуть бути зменшені Замовником залежно від фактичної потреби та реального фінансування видатків. Вимога Замовника щодо зменшення обсягу закупівлі є обов'язковою для Постачальника. У такому разі Сторони вносять відповідні зміни до цього Договору.

1.5. Закупівлю планується провести за загальним фондом КПКВ 2101020, КЕКВ 2210.

**2. ЯКІСТЬ ТОВАРУ ТА ГАРАНТІЯ**

2.1. Постачальник повинен поставити Замовнику Товар вітчизняного або іноземного виробництва (за винятком Товарів виробництва суб'єктів, щодо яких діє рішення про застосування персональних спеціальних економічних та інших обмежувальних заходів (санкцій) відповідно до Закону України "Про санкції"), якість якого повинна відповідати діючим ДСТУ (ISO тощо). В момент передачі (поставки) Товару, Постачальник надає Замовнику відповідні документи, що підтверджують якість Товару або якість сировини з якої виготовлено Товар (паспорт, висновок органу стандартизації, декларація відповідності, сертифікат якості або відповідності тощо).

2.2. Гарантійний термін та термін придатності на продукцію, що поставляється за цим Договором, на умовах дотримання правил використання, зберігання, технічного обслуговування й експлуатації даного виду товару, відповідає чинним вимогам законодавчих та нормативних актів України та обчислюється з дати передачі (прийому) Товару (партії Товару).

2.3. При виявленні Замовником виробничих дефектів Товару (у тому числі прихованих) під час його приймання, Постачальник зобов'язаний протягом 5 (п'яти) календарних днів від дати приймання власними силами і за власний рахунок усунути дефекти Товару, а якщо дефекти усунути не можливо, замінити його на аналогічний якісний Товар.

2.4. У випадку передчасного виходу з ладу або виявлення дефектів Товару (в тому числі прихованих) протягом строку дії гарантії, які не дозволяють продовжити нормальну експлуатацію (**використання**) Товару, за умови що передчасний вихід з ладу або дефекти не є наслідками порушень правил експлуатації та зберігання, Постачальник гарантує безкоштовне усунення дефектів або заміну Товару на якісний протягом 10 (десяти) календарних днів від дати отримання письмового повідомлення Замовника про виявлені недоліки, шляхом заміни неякісного Товару на Товар належної якості. Доставка Товару, що підлягає гарантійному обслуговуванню або заміні, здійснюється Постачальником власними силами і за свій рахунок. У разі усунення дефектів Товару, на який встановлено гарантійний строк експлуатації, цей строк продовжується на час, протягом якого він не використовувався через дефект, а при заміні виробу гарантійний строк обчислюється знову, починаючи із дня заміни.

2.5. Наявність недоліків (дефектів) і строки їх усунення фіксуються двостороннім актом Постачальника і Замовника.

