

Extraction of Pharmaceutical Data from Medical Prescriptions Using Optical Character Recognition (OCR) Model

Suraj Khod¹, Jayant Patil², Prathamesh Patil³, Dr. Manisha Mali⁴

^{1,2,3,4}Dept. of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, India

Abstract:

This study focuses on effectively adding a layer of abstraction while searching for generic medicines or supplements from prescriptions. We leverage the use of AI which demonstrates effective extraction of all the necessary medicinal information from the intricate prescriptions that most users find hard to read or correlate. By utilizing an Optical Character Recognition (OCR) model we solve this problem. The model is designed to be able to extract all the text information provided in the prescription first and apply some parameters that can filter personal information. It also filters harmful or severe medication materials if mentioned; and warns the user. To provide relevant information about the medications or drugs, the system relies on a trusted API endpoint to display precautions, dosage, sources and other important parameters to better assist users regarding their prescription. This reduces the effort of manually trying to understand details behind a medicine and searching countless sites to look for exactly what the user wants. We developed an automated pipeline using OCR and Large Language Models (LLM). The primary goal of this system is to identify the medicines prescribed to a patient accurately, reducing manual data entry and enhancing the efficiency of medical records processing in healthcare settings.

Keywords: OCR; Text Extraction; Open FDA; Paddle OCR;

1. INTRODUCTION

Automation has the potential to significantly impact a number of industries, including healthcare, particularly in the areas of managing prescription drug access and management. Medical prescriptions in the modern healthcare setting sometimes contain complex information that patients, especially the elderly, may find challenging to understand. Intricate prescriptions often don't provide much information about the medicines being prescribed, they highly rely on the fact that the user is most likely to get assistance from a pharmacist while buying medicines from prescription. But it may happen that users cannot get an assistant and need to figure out the prescription themselves for general medicines. To provide this autonomy to patients we try to implement an Optical Character Recognition (OCR) model to address this issue.

Here, we present an automated approach that uses the high-performance OCR framework PaddleOCR to extract text from printed medical prescriptions [6]. After extracting the text, we use Gemini to remove any personal information and flags potentially dangerous drugs, alerting users to seek medical attention if needed. The OpenFDA[6] API, a public database that offers comprehensive information about

medications, including dosages, possible adverse effects, and precautions, is also connected with the system.

By this technique users significantly enhance their ability to gather information about medicines. It reduces the amount of human effort required to decipher printed prescriptions and provides users with reliable, timely, and accurate information about their medicines. Through the integration of OpenFDA, the system offers extensive insights, reducing the possibility of misuse or misunderstanding on the part of the user.

2. IMPLEMENTATION OF AN OCR MODEL

A. General view of OCR

OCR is a technique used to recognize the textual data in non textual file types like images and videos [1]. OCR is generally used to recognize handwritten text, printed text, and text in the wild. With the help of OCR we can easily recognize the prescription data and process it in milliseconds [1].

In the context of this study, OCR's role becomes pivotal when dealing with prescriptions. Most of the medical prescriptions contain a mix of printed and handwritten information, both of which are critical for the accurate interpretation of medications. The OCR model not only needs to be fast, but it also needs to be highly precise in contexts where even minor mistakes could lead to serious health risks.

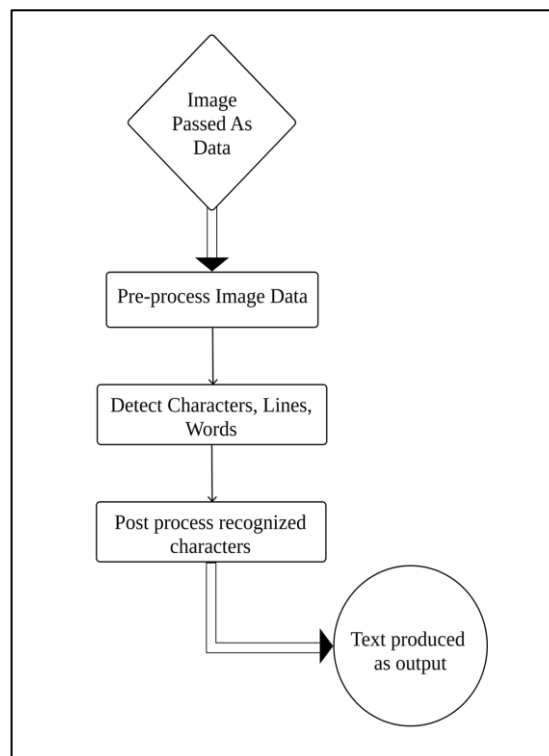


Fig. 1. General workflow diagram of OCR

B. Paddle OCR

PaddleOCR is an open-source OCR tool developed by Baidu's PaddlePaddle deep learning framework [4]. It is designed in a way to recognize and extract text from digital media. OCR supports multiple languages like English, Chinese, Korean, Japanese, and many more [4]. This makes it the best suited tool to scan and extract information independent of language.

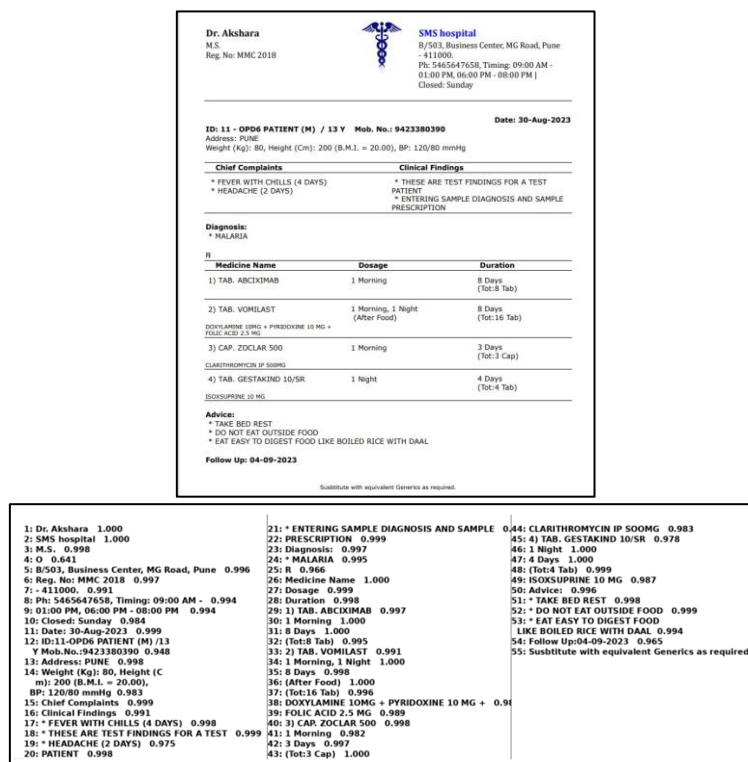
For this particular use case, we need a system that can be made easily accessible throughout multiple communication devices. PaddleOCR solves this issue through its support for multiple deployment

platforms like Python, C++, and Java, and can be deployed on mobile, server, or cloud environments. Since we are targeting to extract only the pharmaceutical text data from input, we can do so using PaddleOCR’s Customizable and Extensible feature where we can fine-tune PaddleOCR models on our own datasets and customize the OCR pipeline according to the needs.

C. Text Extraction Using PaddleOCR

The first step in the pipeline involves extracting text from an image of the medical prescription. For this, we utilized PaddleOCR. Before passing the image to PaddleOCR, we apply image preprocessing techniques to enhance text recognition accuracy. These include converting the image to grayscale, noise reduction, contrast enhancement, and in some cases, deskewing the image to correct for any distortions in the document [5]. These preprocessing steps ensure that the OCR system accurately captures the text, even in low-quality images or those with handwritten content. After preprocessing, PaddleOCR generates both the textual content and bounding box information to localize the text within the image, which can be useful for advanced processing.

For example, we have preprocessed a royalty free prescription image available on the internet. Next we pass it through Paddle OCR and get the extracted text data.



Dr. Akshara
M.S.
Reg. No: MMC 2018

SMS hospital
B/503, Business Center, MG Road, Pune
-411000.
Ph: 5465647658, Timing: 09:00 AM -
01:00 PM, 06:00 PM - 08:00 PM |
Closed: Sunday

ID: 11 - OPDS PATIENT (M) / 13 Y Mob. No.: 9423380390 Date: 30-Aug-2023
Address: PUNE
Weight (Kg): 80, Height (Cm): 200 (B.M.I. = 20.00), BP: 120/80 mmHg

Chief Complaints	Clinical Findings
* FEVER WITH CHILLS (4 DAYS) * HEADACHE (2 DAYS)	* THESE ARE TEST FINDINGS FOR A TEST PATIENT * ENTERING SAMPLE DIAGNOSIS AND SAMPLE PRESCRIPTION

Diagnosis:
* MALARIA

Medicine Name	Dosage	Duration
1) TAB. ABCIXIMAB	1 Morning	8 Days (Tot: 8 Tab)
2) TAB. VOMILAST	1 Morning, 1 Night (After Food)	8 Days (Tot: 16 Tab)
3) CAP. ZOCLAR 500	1 Morning	3 Days (Tot: 3 Cap)
4) TAB. GESTAKIND 10/SR	1 Night	4 Days (Tot: 4 Tab)

Advice:
* TAKE BED REST
* DO NOT EAT OUTSIDE FOOD
* EAT EASY TO DIGEST FOOD LIKE BOILED RICE WITH DAAL

Follow Up: 04-09-2023

Substitute with equivalent Generics as required.

1: Dr. Akshara 1.000	21: * ENTERING SAMPLE DIAGNOSIS AND SAMPLE PRESCRIPTION 0.999	044: CLARITHROMYCIN IP SOOMG 0.983
2: SMS hospital 1.000	22: Diagnosis: 0.997	45: 4) TAB. GESTAKIND 10/SR 0.978
3: M.S. 0.998	24: * MALARIA 0.995	46: 1 Night 1.000
4: O 0.641	25: R 0.966	47: 4 Days 1.000
5: B/503, Business Center, MG Road, Pune 0.996	26: Medicine Name 1.000	48: (Tot: 4 Tab) 0.999
6: Reg. No: MMC 2018 0.997	27: Dosage 0.999	49: ISOSUPRINE 10 MG 0.987
7: -411000, 0.991	28: Duration 0.998	50: Advice: 0.996
8: Ph: 5465647658, Timing: 09:00 AM - 0.994	29: 1) TAB. ABCIXIMAB 0.997	51: * TAKE BED REST 0.998
9: 01:00 PM, 06:00 PM - 08:00 PM 0.994	30: 1 Morning 1.000	52: * DO NOT EAT OUTSIDE FOOD 0.999
10: Closed: Sunday 0.984	31: 8 Days 1.000	53: * EAT EASY TO DIGEST FOOD LIKE BOILED RICE WITH DAAL 0.994
11: Date: 30-Aug-2023 0.999	32: (Tot: 8 Tab) 0.995	54: Follow Up: 04-09-2023 0.965
12: ID: 11 - OPDS PATIENT (M) / 13 Y Mob. No.: 9423380390 0.948	33: 2) TAB. VOMILAST 0.991	55: Substitute with equivalent Generics as required
13: Address: PUNE 0.998	34: 1 Morning, 1 Night 1.000	
14: Weight (Kg): 80, Height (Cm): 200 (B.M.I. = 20.00), BP: 120/80 mmHg 0.983	35: 8 Days 0.998	
15: Chief Complaints 0.999	36: (After Food) 1.000	
16: Clinical Findings 0.991	37: (Tot: 16 Tab) 0.996	
17: * FEVER WITH CHILLS (4 DAYS) 0.998	38: DOXYLAMINE 10MG + PYRIDOXINE 10 MG + FOLIC ACID 2.5 MG 0.989	
18: * THESE ARE TEST FINDINGS FOR A TEST PATIENT 0.999	39: 3) CAP. ZOCLAR 500 0.998	
19: * HEADACHE (2 DAYS) 0.975	40: 1 Morning 0.982	
20: PATIENT 0.998	41: 3 Days 0.997	
	42: 3 Days 0.997	
	43: (Tot: 3 Cap) 1.000	

Fig. 2. Example prescription output from PaddleOCR

3. PREPROCESSING ON EXTRACTED TEXT

Once the text is extracted, we move to the preprocessing stage, which plays a crucial role in refining the raw output from the OCR. Medical prescriptions often contain noisy or irrelevant information, such as doctor's instructions, dosage information, dates, and patient names, which are not pertinent to the task of medicine extraction.

To handle this, the extracted text undergoes several preprocessing steps:

a. Noise Removal

The goal is to clean the text output from the OCR model by eliminating irrelevant characters, symbols, and artifacts that may have been mistakenly recognized due to the quality of the image or errors inherent in the recognition process.

TABLE I.

Type of noise	Reasons for Occurrence	Specimens	Techniques of removal
Character-level Noise	i) Misrecognition of special symbols. ii) Incorrect characters that don't belong to the actual text	'@' or '#'	Regular Expressions
Formatting Artifacts	Line breaks, spaces, and indentation.	'\n' escape character or new breaking space	Text Normalization
Non-essential Text	i) Irrelevant information like dates. ii) Patient details. iii) Doctor's instructions	'30-05-2023' or 'Mr/Ms' or 'prescription context'	NLP, Contextual Information

Fig. 3. Refer [1] Noise category table

b. Regular Expression-Based Filtering

Regular expressions are a common approach to filtering out unwanted characters and patterns from the OCR output. For example, by defining patterns to recognize only alphanumeric characters (letters and numbers) or certain medical terms, we can filter out special symbols or irrelevant sections. This method is flexible and allows customization to remove specific types of noise (e.g., remove hashtags or irrelevant punctuation marks) [7].

```
# Remove special characters
clean_text = re.sub(r'^a-zA-Z0-9\s|', "", raw_text)
```

Fig. 4. Regular Expression to remove special characters

c. Text Segmentation

The extracted text is further segmented based on common patterns found in medical prescriptions, which allows us to distinguish between different sections such as the diagnosis, medication list, dosage instructions, and general notes. This can be particularly useful for identifying sections of text that OCR might otherwise misclassify [7].

4. MEDICINE NAME EXTRACTION USING LARGE LANGUAGE MODELS (LLMs)

After preprocessing, the cleaned text is passed into a Large Language Model (LLM) for further analysis. The LLM used in our pipeline is fine-tuned specifically for medical text and is trained to recognize and extract medicine names from the processed text. LLMs, such as GPT and Llama, are particularly powerful in this context because of their ability to understand context and semantics, allowing them to differentiate between drug names and other information that might be presented in the prescription, such as treatment instructions or patient information.

Several enhancements are made to ensure accurate medicine identification:

a. Domain-Specific Fine-Tuning

The LLM is fine-tuned using a domain-specific corpus that includes medical terminology, prescription formats, and common drug names. This enhances the model's ability to recognize medicines, even in the presence of noise or minor OCR errors. One significant advantage of domain-specific fine-tuning is that the LLM becomes more resilient to noise or minor OCR errors, such as slight misrecognitions in handwriting or printed text. In healthcare, where prescriptions often contain handwritten components, the LLM's ability to interpret text with partial errors or irregularities is crucial. By understanding context more effectively, the fine-tuned model can still correctly identify drug names even if the surrounding text is unclear or distorted [12]. This allows for high accuracy in recognizing medicines and reduces the number of false positives in the identification process.

b. Named Entity Recognition (NER)

An NER module is integrated into the LLM pipeline to identify specific entities such as drug names. This ensures that only relevant entities—medications—are extracted while ignoring other named entities such as doctor names or places [12]. Drug names are just one of many types of named entities that could be mentioned. However, not all of this information is necessary when the goal is to identify medications for further processing. The NER model works by categorizing the text into specific classes, such as medication, dosage, doctor, and patient information [12]. It uses predefined medical taxonomies and dictionaries, as well as context learned from the domain-specific corpus, to efficiently separate medications from other entities.

This filtering mechanism enhances accuracy, ensuring that the system doesn't confuse medications with other entities [13]. For example, if a prescription mentions "Dr. Smith" and "Tylenol," the NER model will recognize "Dr. Smith" as a person and "Tylenol" as a medication, preventing any misclassification. The precision of NER is critical in preventing the model from making incorrect predictions and ensures that the focus remains on the relevant parts of the prescription.

c. Drug Synonym Handling and Correction

The LLM is also equipped with a mechanism for handling drug synonyms and common misspellings. This ensures that variants of the same drug (e.g., "Tylenol" and "Acetaminophen") are correctly recognized and standardized [13]. Additionally, spell-checking mechanisms correct OCR-induced errors in drug names, which is particularly important for handwritten prescriptions. This mechanism relies on a comprehensive database of drug synonyms and common misspellings [10]. When a drug name is identified, the LLM compares it to this database to determine if it is a variant of a known medication. For example, if "Acetaminofen" is recognized in the prescription due to an OCR-induced misspelling, the system can map this to the correct drug name "Acetaminophen." Similarly, brand-to-generic name mappings are handled, ensuring that both "Tylenol" and "Acetaminophen" are recognized as the same substance [10].

By standardizing drug names, the LLM ensures consistency in the output and reduces ambiguity for end users. This feature is particularly useful when patients might not be familiar with the generic names of drugs listed in their prescriptions. Additionally, the correction mechanism improves the reliability of the system by automatically fixing minor spelling errors that could have been introduced by OCR inaccuracies [8].

5. POST-PROCESSING AND VALIDATION

Once the LLM extracts potential medicine names from the text, we apply a post-processing phase to validate and refine the results:

a. Cross-Validation with Drug Databases

Just fetching the medicines metadata isn't much helpful to the users. To provide a better and purposeful service we check the availability of the medicines in an inventory. The extracted medicine names are cross-referenced with authoritative drug databases such as RxNorm or DrugBank [10]. This ensures that the identified drugs are valid and correctly spelled. If any discrepancies are found (e.g., unrecognized drug names), the system either flags them for manual review or attempts to correct them based on database lookups. Furthermore if a user intends to place an order for the medicines they can place an order for them. The cross-validation also makes sure that the medicines are available for delivery.

b. Handling Abbreviations and Dosages

For cases such as abbreviated drug names or combined dosage instructions (e.g., "Paracetamol 500mg"), the post-processing step can separate the drug name from the dosage instructions, This allows the system to extract only the relevant medication names [3]. For more complex dosage instructions such as 'Paracetamol 500mg TID' or 'Ibuprofen 400mg q6h PRN'. Thus, the model should be able to separate that In addition to separating the drug name from the dosage. The post-processing system recognizes typical medical abbreviations associated with dosing frequency, such as TID, BID, and PRN [3]. This preserves the entire context of the drug's intended dosage for the system, which may be crucial for users who require thorough medication information.

6. MODEL EVALUATION AND ACCURACY ASSESSMENT

To assess the performance of the proposed system, we conducted experiments using a test dataset of prescription images collected from real-world healthcare environments. Several evaluation metrics were used, including precision, recall, and F1-score, to measure the accuracy of medicine name extraction [11]. Additionally, the system was tested for robustness in handling handwritten prescriptions, varying image qualities, and multi-language prescriptions, as these are common challenges in real-world healthcare scenarios.

7. POST EXTRACTION INFORMATION GATHERING

After extraction of text data (especially drug names) we pass this to openFDA API, these names are sent as query parameters to openFDA's Drug API endpoints. Endpoints like /drug/label, /drug/event, and /drug/enforcement are provided by OpenFDA. The /drug/label endpoint is generally used to access generic data about a medication, such as its uses, adverse effects, and active ingredients [6].

```
https://api.fda.gov/drug/label.json?s
```

Fig. 5. Structure of query to OpenFDA

Here, the extracted drug name (such as "Paracetamol" or "Ibuprofen") replaces the placeholder drug_name in the query. The OpenFDA API then attempts to find a precise match for the medication in its database. If the drug name is identified in its records, OpenFDA retrieves detailed metadata related to that drug. If no match is found, the API returns an empty response, signaling that the drug is either unlisted or the query may need adjustment (e.g., trying the generic name instead of the brand name) [6].

The /drug/label endpoint is one of the most commonly used in the pipeline because it provides extensive

general information about medications [6]. This endpoint returns data related to:

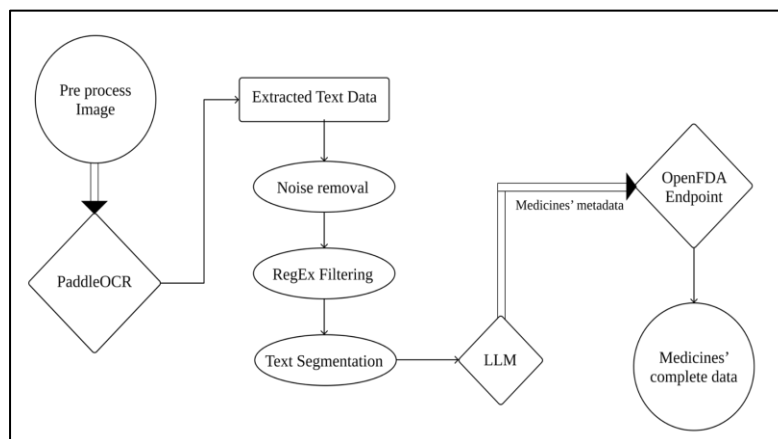
1. **Purpose and Indications:** The primary uses of the drug, including what conditions it is meant to treat.
2. **Dosage and Administration:** Recommended dosages and instructions for administering the medication, including specific details for different age groups or conditions.
3. **Warnings and Precautions:** Important safety information, such as potential risks, interactions with other drugs, and contraindications.
4. **Side Effects:** Known adverse reactions or side effects that have been reported during clinical trials or post-market surveillance.
5. **Active Ingredients:** A list of the active components in the drug, which is useful for comparing different formulations or generic versions.
6. **Manufacturer Information:** Details about the drug's manufacturer, including the company name and contact details.

8. PROPOSED METHODOLOGY

Optical Character Recognition (OCR) Model Implementation is done using PaddleOCR Setup. We implement the PaddleOCR framework for efficient text extraction from printed prescription images. Text Extraction Process uses PaddleOCR to convert images to machine-readable text. Structure the extracted text into fields for each prescription component (medicine name, dosage, instructions). Error handling and correction algorithms are implemented to handle OCR misinterpretations or ambiguities, refining text recognition accuracy.

We ensure data filtering and sanitization through NLP methods (e.g., entity recognition) to identify and redact any personal identifiers, maintaining patient privacy. Medication safety check is done using a rule-based or model-based filtering system to flag dangerous or restricted drugs within the extracted data. Raise alerts for potentially harmful medications.

We Connect with OpenFDA to fetch detailed information on identified medications, including dosage, side effects, and precautions. Matched and extracted medicine names from OpenFDA records are considered as relevant metadata with information such as warnings and recommended usage to the user. Text processing pipeline is automated with a workflow that integrates each stage—from OCR to data filtering and enrichment. Measure OCR accuracy, filtering precision, and API matching reliability using metrics like F1-score and recall.



Flowchart of proposed methodology to implement OCR for fetching medicines metadata

demonstrated the viability of automating pharmaceutical data extraction from medical prescriptions. This approach not only simplifies the process for end-users but also ensures reliable and accurate results, enhancing the overall healthcare experience. Further improvements could involve enhancing OCR accuracy for handwritten prescriptions and expanding the system's language support for non-English medical terms. This effectively bridges the gap between understanding the intricate medical prescription and provides an easily understandable explanation and metadata fetch from the OpenFDA API endpoints. This solution offers a better user experience, effective method of interpreting prescription data and helping e-commerce businesses to provide their service to larger audiences. An abstract model of 'add to cart' feature can be implemented upon this solution to add all the medications directly from prescription instead of letting users search and try or fail to understand medical terms. This solution also lowers medication errors and enhances patient safety. This technology is extremely beneficial not just for patients but also for healthcare practitioners, who can use it to streamline processes and lessen administrative responsibilities due to its precise and dependable outcomes.

REFERENCES

1. A. Singh, S. Jangra and G. Aggarwal, "EnvisionText: Enhancing Text Recognition Accuracy through OCR Extraction and NLP-based Correction," 2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2024, pp. 47-52, doi: 10.1109/Confluence60223.2024.10463478. J. Shin, "Investigating the accuracy of the openFDA API using the FDA Adverse Event Reporting System (FAERS)," 2014 IEEE
2. B. K. Pattanayak, A. K. Biswal, S. R. Laha, S. Pattnaik, B. B. Dash and S. S. Patra, "A Novel Technique for Handwritten Text Recognition Using Easy OCR," 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), Erode, India, 2023, pp. 1115-1119, doi: 10.1109/ICSSAS57918.2023.10331704.
3. C. C. Paglinawan, M. Hannah M. Caliolio and J. B. Frias, "Medicine Classification Using YOLOv4 and Tesseract OCR," 2023 15th International Conference on Computer and Automation Engineering (ICCAE), Sydney, Australia, 2023, pp. 260-263, doi: 10.1109/ICCAE56788.2023.10111387
4. Du, Yuning, et al. "Pp-ocr: A practical ultra lightweight ocr system." arXiv preprint arXiv:2009.09941 (2020).
5. International Conference on Big Data (Big Data), Washington, DC, USA, 2014, pp. 48-53, doi: 10.1109/BigData.2014.7004412.
6. J. Shin, "Investigating the accuracy of the openFDA API using the FDA Adverse Event Reporting System (FAERS)," 2014 IEEE
7. Kathuria, Abhinav, Anu Gupta, and R. K. Singla. "A review of tools and techniques for preprocessing of textual data." Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 1 (2021): 407-422
8. K. V. Ujwal Karanth, A. T. Sujana, Y. R. Thanay Kumar, S. Joshi, K. P. Asha Rani and S. Gowrishankar, "Breaking Barriers in Text Analysis: Leveraging Lightweight OCR and Innovative Technologies for Efficient Text Analysis," 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2023, pp. 359-366, doi: 10.1109/ICACRS58579.2023.10404305.
9. Mittal, Rishabh, and Anchal Garg. "Text extraction using OCR: a systematic review." 2020 second international conference on inventive research in computing applications (ICIRCA). IEEE, 2020.

10. N. Maitrichit and N. Hnoohom, "Intelligent Medicine Identification System Using a Combination of Image Recognition and Optical Character Recognition," 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), Bangkok, Thailand, 2020, pp. 1-5, doi: 10.1109/iSAI-NLP51646.2020.9376816.
11. Singh, Aditya, Sahil Jangra, and Garima Aggarwal. "EnvisionText: Enhancing Text Recognition Accuracy through OCR Extraction and NLP-based Correction." 2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE, 2024.
12. S. Jasmine, R. Ch and R. Srikavya, "Medicine Drug Name Detection Object Recognition using Deep Learning based OCR System," 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2023, pp. 1-6, doi: 10.1109/ICIICS59993.2023.10421226.
13. Z. Wang, Z. Gong and Z. Pei, "Research on Intelligent Text Detection Based on Embedded OCR," 2023 2nd International Conference on Artificial Intelligence and Computer Information Technology (AICIT), Yichang, China, 2023, pp. 1-4, doi: 10.1109/AICIT59054.2023.10277728.