INDUSTRIAL TRAINING REPORT

On the Topic

“Extraction and Classification of Images based on Caption from a Well Construction Report PDF”

At

GEOPIC CENTRE

OIL & NATURAL GAS CORPORATION

DEHRADUN

SUBMITTED BY –

Mohammad Suhail

Pursuing B.Tech from
Department of Computer Engineering
Zakir Hussain College of E & T,
Aligarh Muslim University, Aligarh

UNDER GUIDENCE OF –

Mr. Sanjay Chakravorty

Dy. General Manager (Programming),
Geopic Centre,
ONGC, Dehradun

DECLARATION

I hereby declare that the work presented in this Project Report entitled **"Extraction and Classification of Images based on Caption from a Well Construction Report PDF"**, done as a Part of Industrial Training at Geopic Centre, Oil & Gas Corporation, Dehradun, is an authentic record of my work carried out during the Industrial Training from 1st January 2025 to 30th February 2025, under the guidance of **Mr. Sanjay Chakravorty**, General Manager (PROG.), National Database, ONGC, Dehradun.

Suhail

(Mohammad Suhail)

CERTIFICATE

This is to certify that **Mr. Mohammad Suhail**, a student of B.Tech (Computer Engineering) of Zakir Husain College of Engineering and Technology, Aligarh Muslim University had undergone an Industrial Training at EPINET, KDMIPE Campus, ONGC Dehradun. The project work entitled **“Extraction and Classification of Images based on Caption from a Well Construction Report PDF”**, embodies the original work done by Mr. Mohammad Suhail during his Two-month Industrial Training period from 1st January 2025 to 30th February 2025.

Mr. Sanjay Chakravorty

Dy. General Manager (Programming)
National Database,
ONGC, Dehradun

ACKNOWLEDGEMENT

My internship with Oil and Natural Gas Corporation Ltd. was a journey of profound learning and immense professional growth. I consider myself exceptionally fortunate to have been granted the privilege to be a part of such an extraordinary opportunity, which has undeniably left an indelible mark on my personal and professional journey.

I am deeply grateful to Mr. Sanjay Chakravorty, DGM (Programming) at National Database, ONGC Dehradun, who graciously undertook the role of my mentor during this transformative phase. His insightful advice and unwavering support illuminated my path throughout this immersive journey.

This internship marks a crucial step in my career journey, equipping me with invaluable skills and knowledge of immense potential. I'm wholeheartedly committed to applying these assets effectively and am determined to realize my career aspirations through continuous growth.

I eagerly anticipate ongoing collaboration with the exceptional individuals I've met, believing in our ability to achieve remarkable feats together. In conclusion, I'm profoundly grateful to my parents for their unwavering support, which has been my driving force throughout this endeavour.

Sincerely
Mohammad Suhail

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

ABOUT ONGC

Oil and Natural Gas Corporation Limited (ONGC) is a major Indian multinational oil and gas company headquartered in Dehradun, Uttarakhand, India. Established in 1956, ONGC is a flagship organization under the Ministry of Petroleum and Natural Gas of the Government of India.

ONGC is engaged in the exploration, production, and refining of oil and gas, making it one of the largest publicly-traded oil and gas companies in India. The company's activities span the entire hydrocarbon value chain, from exploration to production, refining, and distribution. ONGC plays a vital role in ensuring the energy security of India by contributing significantly to the country's domestic oil and gas production.

HISTORY

Established in the late 1950s, the Oil and Natural Gas Corporation Limited (ONGC) has carved a storied history as a vital pillar of India's energy landscape. Born from the vision of post-independence self-sufficiency, ONGC was founded in 1955 to spearhead the exploration and production of indigenous oil and gas resources. Rapidly making its mark, ONGC achieved its first significant oil discovery in 1960 and extended its reach to offshore drilling, discovering the Bombay High oil field. The subsequent decades witnessed ONGC's evolution into a full-fledged corporation, marked by its forays into international ventures, adoption of advanced technologies, and integration of various oil-related entities. Amidst growing environmental awareness, ONGC embraced sustainable practices and ventured into renewable energy. Today, its journey stands as a testament to its enduring commitment to powering India's progress, innovating for a sustainable future, and maintaining a pivotal role in the nation's economic and energy security endeavours.

VISION AND MISSION

VISION

To be global leader in integrated energy business through sustainable growth, knowledge excellence and exemplary governance practices.

MISSION

World Class

- Dedicated to excellence by leveraging competitive advantages in R&D and technology with involved people.
- Imbibe high standards of business ethics and organizational values.
- Abiding commitment to safety, health and environment to enrich quality of community life.
- Foster a culture of trust, openness and mutual concern to make working a stimulating and challenging experience for our people.
- Strive for customer delight through quality products and services.

Integrated in Energy Business

- Focus on domestic and international oil and gas exploration and production business opportunities.
- Provide value linkages in other sectors of energy business.
- Create growth opportunities and maximize shareholder value.

FRONTIERS OF TECHNOLOGY

- **Seismic Excellence:** ONGC excels in seismic data handling, using advanced acquisition, processing, and interpretation facilities.
- **Virtual Reality Prowess:** Featuring a world-ranking Virtual Reality Interpretation setup, ONGC is committed to immersive data analysis.
- **Global Partnerships:** Collaborations with major industry players like Transocean, Schlumberger, and more, emphasize ONGC's global impact.
- **International Reach:** Alliances with Petro bras, Norsk, ENI, and Shell highlight ONGC's significance in the global energy sector.
- **Efficiency Emphasis:** ONGC's extensive ERP implementation showcases its dedication to operational efficiency and effective management.

GLOBAL RANKING

- **Top Energy Company in India:** Oil and Natural Gas Corporation (ONGC) ranked as the best energy company in India in the prestigious Platts Top 250 Global Energy Company Rankings 2014.
- **Global Recognition:** ONGC improved its global ranking, securing the 21st position among global energy majors, and climbed to the 5th spot in the Asia/Pacific (APAC) region, up from 7th the previous year.
- **Exploration and Production Leadership:** ONGC maintained its 3rd position as a global Exploration and Production company, further solidifying its industry leadership. • Fast-Lane Initiatives: ONGC's upcoming initiatives and accelerated production plans are set to contribute to even greater milestones in the future.
- **BT 500 Ranking:** ONGC's recent rise to Number 2 among Indian corporate giants in the BT 500 ranking adds to its string of achievements.
- **Asian Influence:** ONGC's ranking at 5 in the APAC region reflects the growing influence of Asian companies in the global energy sector.
- **Indian Companies in the List:** Thirteen Indian companies, including Reliance Industries Ltd, Indian Oil Corporation, and NTPC, are featured in the Top 250 Global Energy Company list by Platts.
- **Performance Metrics:** The Platts survey evaluates companies based on key financial metrics such as asset worth, revenues, profits, and return on invested capital. All companies in the list have assets exceeding US $5 billion.

EDS (Formerly EPINET)

About

Exploration Data Services (EDS) comes under an umbrella body Exploration Digital Space. During past five decades of extensive hydrocarbon exploration over a vast geographical area covering almost all the sedimentary basins in India, ONGC has acquired enormous volume of geological, geophysical and engineering data. In line with the global upstream oil majors, underlining the importance of computerized data archival, ONGC also initiated data management activities way back in 1976. A well information system on IBM-370 at KDMIPE was developed "in-house". The data management activities got a boost when VAX 3400 computer system along with RDBMS software was commissioned at KDMIPE in 1991. Furthermore, other isolated systems also emerged at different work centres of ONGC to cater to the data management needs. These in-house database systems were developed using different types of formats and platforms. Eventually, in-house database systems were felt inadequate and required to be replaced with system matching international standards in the industry. ONGC, in late 1990's decided to implement the recent advancements of the web based E&P information technology into its upcoming data management systems. The new implementation plan aimed to make available all information pertaining to an area, including the work previously done by archiving historical analysis, to business users.

In the aforesaid background, ONGC decided to implement **Exploration and Production Information Network (EPINET)** project comprising of people, processes, tools, data and a hierarchy of corporate, regional and working project database in a phased manner. The project was intended to establish an organization wide dynamic, "Virtual" database having GIS features and Web 2 capabilities, to interconnect different data stores located at geographically diversified locations. Having taken shape according to the vision, intention and need of various work-centres of ONGC, today, EPINET is ONGC's full-fledged technical data repository housing all technical data generated throughout ONGC. This includes WCRs, Log data, Seismic data, Geology Data, Lab data, Drilling Data and Production Data.

SOFTWARES –

FINDER

Until 2013, EPINET housed all its data using software called “Finder”. Finder is a product of Schlumberger Information Systems (SIS). Finder uses Oracle Database and is built to run on the old Sun Operating system. A Sun server is used to host the Finder software.

THREE-TIER SETUP

When Finder was in use, EPINET was organized in a three-tier model with secondary and tertiary level data-centres spread across each basin. The following Diagram shows the three-tier setup:

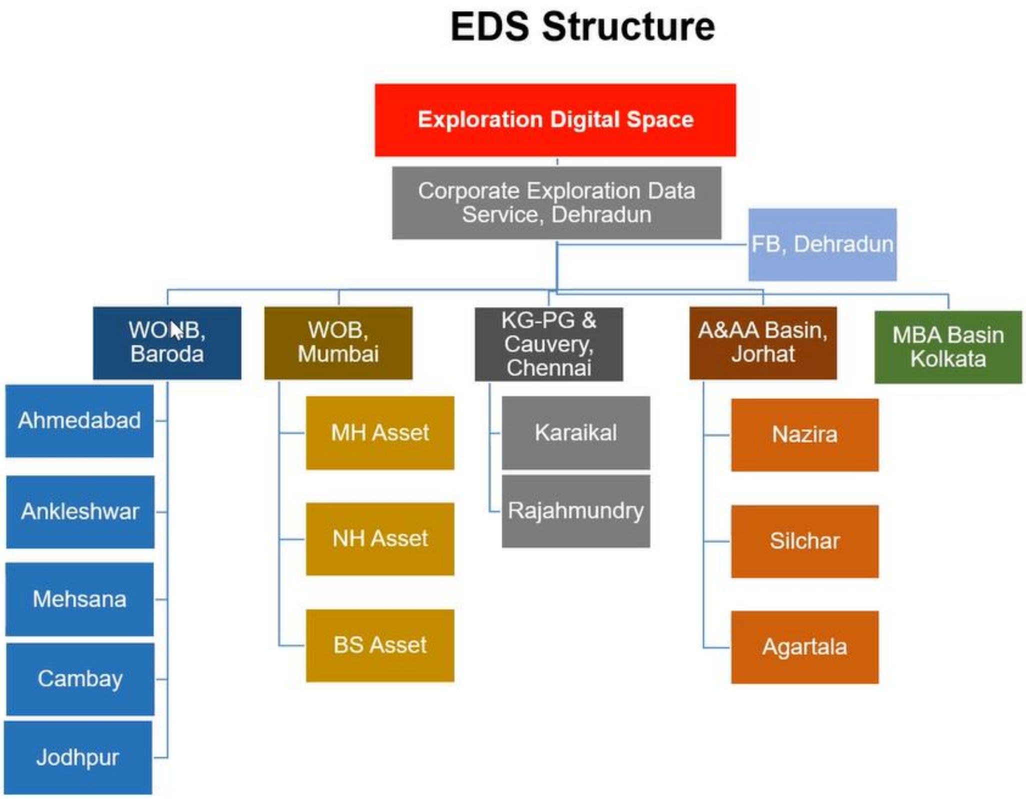


Figure 1: Three Tier Structure of EDS under Finder Software.

ProSource

After 2013, all data in EPINET was managed through a software package called ProSource". ProSource has been in full use only since December 2013. It is also a product of Schlumberger Information Systems (SIS). ProSource uses Oracle Database to manage data and is built on the seabed Data Model. The User Interface of ProSource is web based and is built on Java. ProSource runs on Red Hat Enterprise Linux 5.5. A product of Oracle called "Golden Gate" is used to replicate data from all centres into the database at Dehradun Corporate centre. Migration from Finder to the new ProSource system took places in several phases, covering one region at a time. The final phase completed in December 2013 with the migration of data at Corporate EPINET centre, Dehradun. Below is a screenshot of the new ProSource interface:

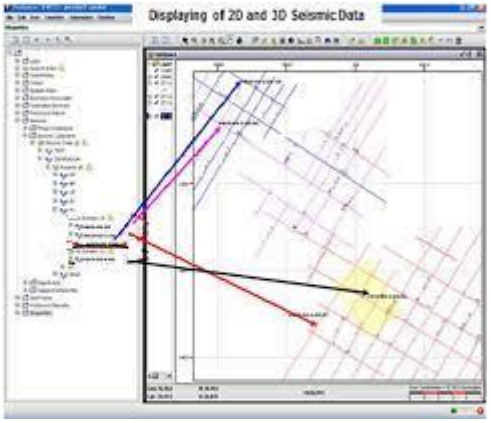


Figure 2: Interface of ProSource Software.

DATABASE CENTERS & DATA FLOW

Under ProSource, EPINET was organized into six data centres in a two-tier model with the corporate centre housed at KDMIPE, Dehradun. The data centres and the regions covered are as follows:

Table 1: Data Centres under ProSource Software

Data Centre Location	Basin / Region covered	Region
Dehradun	Frontier Basin	Northern Region (NR)
Baroda	Western Onshore Basin	Western Region (WR)
Mumbai	Western Offshore	Mumbai Region (MR)
Kolkata	MBA Basin	Central Region (CR)
Jorhat	A&AA Basin	Eastern Region (ER)
Chennai	KG-PG Basin	Southern Region (SR)

Data from respective regions are populated and maintained at the respective centres. Data from all centres are centrally replicated at the Corporate EPINET Centre, Dehradun via Golden Gate, and a plugin of Oracle. The following diagram shows the setup:

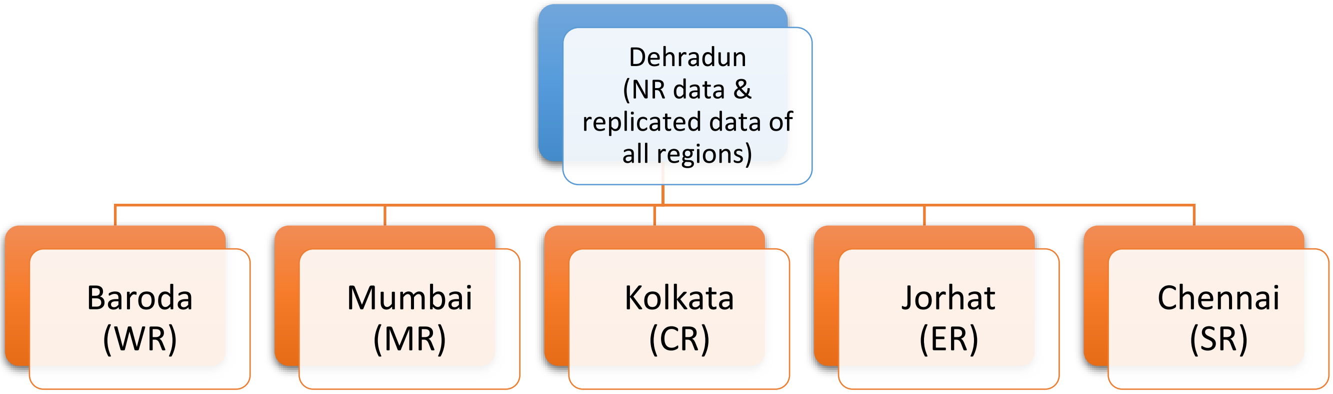


Figure 3: Two-tier setup of EPINET under ProSource.

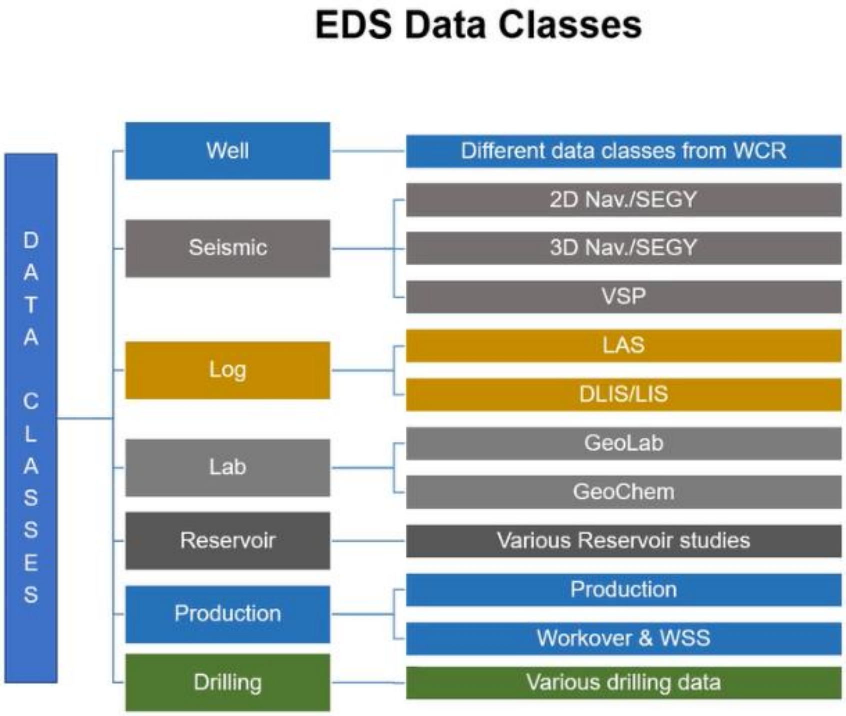
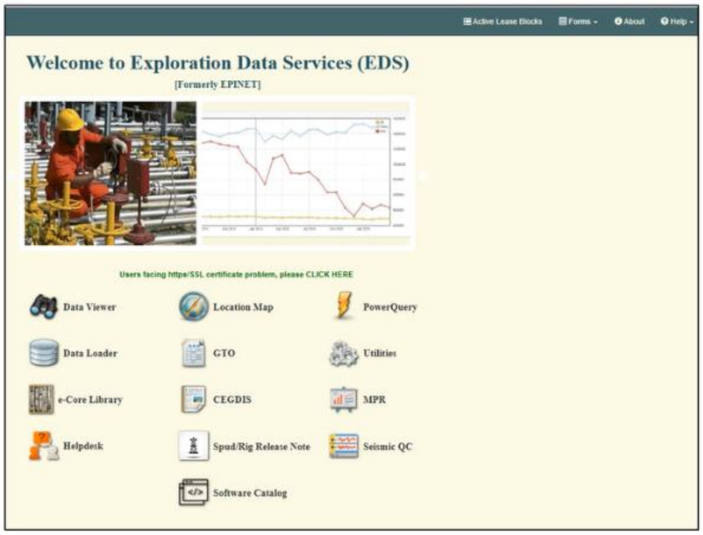


Figure 4: Data Classes under EDS.

DEVELOPMENT OF IN-HOUSE DATA MANAGEMENT SOFTWARE

During November 2012, it was decided that efforts would be made to maintain EPINET system using in-house talent. Eventually, in-house maintenance began in October 2014, saving ₹12 Crore annually since then. Leveraging in-house talent across all EPINET sites and collaborating over a period of 2 years, the Beta version of EPINET Portal & Loaders were ready and was rolled out exclusively for in-house testing by EPINET members, at the Annual Coordination Meeting held at EPINET Chennai. After extensive testing and improvements, in July 2017, the tested and improved version of In-house EPINET Software is rolled out across ONGC by Director (Exploration), Sh. A K Dwivedi via video-conferencing and was jointly witnessed by various offices across ONGC joining the live video conference. This In-house software is currently in use, with new features being added consistently.



Welcome to Exploration Data Services (EDS)

[Formerly EPINET]

Users facing https/SSL certificate problem, please CLICK HERE

Figure 5: EDS Portal – Home Page

INTRODUCTION TO PROJECT

Well completion reports contain critical visual information such as diagrams, charts, and schematics that help in assessing well performance and integrity. However, extracting and classifying these images manually is a time-consuming task. This project aims to automate the extraction of images from well completion report PDFs and classify them into predefined categories using machine learning techniques.

CHAPTER 2

METHODOLOGY

The project follows a structured pipeline for extracting and classifying images from well completion report PDFs. The methodology consists of the following key stages:

2.1 Training a YOLOv8 Model for Image Detection

- A YOLOv8 object detection model is trained to detect images embedded within PDFs.
- The model is trained on a dataset containing various document images, ensuring robustness in detecting different image formats and layouts.
- After training, the model is saved for later inference.

2.2 Extracting Captions Using PyMuPDF (fitz)

- Once an image is detected, its associated caption is identified in the vicinity using fitz from the PyMuPDF library.
- This is achieved by analysing text blocks around the detected image, ensuring accurate caption extraction.

2.3 Dataset Preparation for Classification

- To enhance classification accuracy, a self-written script is used to capture hundreds of captions from multiple PDFs.
- These extracted captions are manually labelled with integer class identifiers, creating a labelled dataset for supervised learning.

2.4 Training the Classification Model

- Multiple machine learning and deep learning models are tested on the labelled caption dataset.
- The model that yields the best classification performance is selected and saved for deployment.

2.5 Final Processing Pipeline

- The final system is designed to process one or more PDFs at a time.
- A file browse popup allows users to select PDFs for processing.

- The pipeline executes the following steps:
  - ✓ Image Detection: The saved YOLOv8 model detects images in the PDF.
  - ✓ Caption Extraction: Captions near detected images are extracted using PyMuPDF.
  - ✓ Classification: The extracted captions are passed through the saved NLP-based classification model, which assigns them to predefined categories.
  - ✓ Storage: The images and their corresponding captions are stored in separate folders based on their classified categories.

This approach ensures an automated, efficient, and accurate extraction and classification of images from well completion reports, improving the accessibility and usability of well log data.

CHAPTER 3

TRAINING YOLO MODEL

3.1 Data Preparation

- Each page of provided PDF is captured and then exported to PNG (Portable Network Graphics) of 300 dpi (dot per inch).

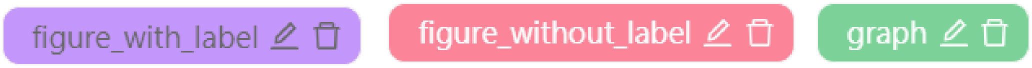
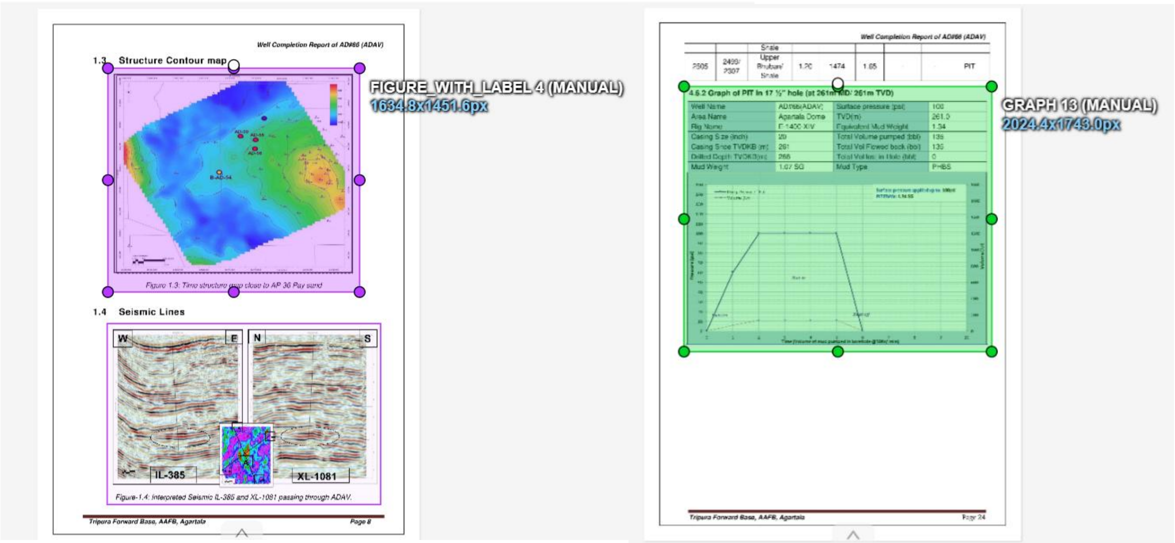


Figure 6: Classes for YOLO Classification (Primary Classes).

- These images are annotated using CVAT tool under 3 categories namely –
  - **a. figure_with_label** – To capture all those images that have captions associated with them below the images.
  - ***b. figure_without_label* –** To capture all those images that are without captions.
  - **c. graph –** To capture all those images that are graphs and lhas the caption given above the placed image.

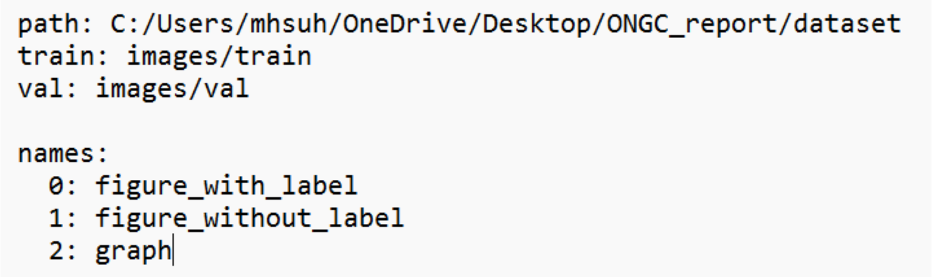


Well Completion Report of AD#86 (ADAV)
1.3 Structure Contour map
Figure 1.3: Time structure map close to AP 30 Pay sand
1.4 Seismic Lines
W
E
N
S
IL-385
XL-1081
Figure-1.4: Interpreted Seismic IL-385 and XL-1081 passing through ADAV.
Tripura Forward Base, AAFB, Agartala
Page 8
FIGURE_WITH_LABEL 4 (MANUAL)
1634.8x1451.6px
Well Completion Report of AD#86 (ADAV)
4.5.2 Graph of PIT in 17 ½" hole (at 261m MD: 261m TVD)
Tripura Forward Base, AAFB, Agartala
Page 24
GRAPH 13 (MANUAL)
2024.4x1743.0px

Figure 7: Image Showing Manual Annotation in CVAT.

- After annotation all the images manually, these are exported to YOLO supported version in which labels are text files containing class index and coordinates of left bottom vertex and right top vertex.

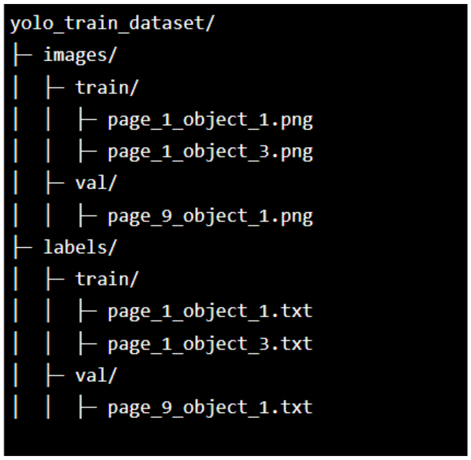
- YAML file is created, that shall be used to configure the Yolo Model, so config.yaml is created.



```
path: C:/Users/mhsuh/OneDrive/Desktop/ONGC_report/dataset
train: images/train
val: images/val

names:
  0: figure_with_label
  1: figure_without_label
  2: graph
```

Figure 8: YAML Configuration file for YOLO.

- The Folder structure of Dataset must strictly follow the specified structure.



```
yolo_train_dataset/
├ images/
│  ├ train/
│  │  ├ page_1_object_1.png
│  │  ├ page_1_object_3.png
│  ├ val/
│  │  ├ page_9_object_1.png
├ labels/
│  ├ train/
│  │  ├ page_1_object_1.txt
│  │  ├ page_1_object_3.txt
│  ├ val/
│  │  ├ page_9_object_1.txt
```

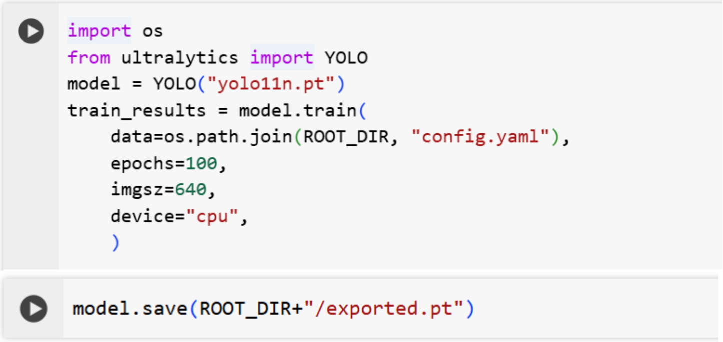
Figure 9: Directory Structure of Dataset for YOLO.

3.2 Model Selection

- The YOLOv8n (nano version) was selected for training due to its balance of speed and accuracy.
- The model contains 238 layers, 2,582,737 parameters, and 6.3 GFLOPs.

3.3 Model Training

1. The model was trained using the Ultralytics YOLOv8 framework.
2. Hyper parameters used:
    a. Epochs: 100
    b. Batch size: 16
    c. Image size: 640x6402/2
    d. Optimizer: Adam
    e. Learning rate: 0.001
    f. Data augmentation: Mosaic augmentation, blur, CLAHE, grayscale conversion.
    g. Early stopping: Not used, but monitored validation loss manually.
3. The model was trained on Google Colab on CPU (Intel Xeon 2.20GHz).
4. Model is saved for future use.



```python
import os
from ultralytics import YOLO
model = YOLO("yolo11n.pt")
train_results = model.train(
    data=os.path.join(ROOT_DIR, "config.yaml"),
    epochs=100,
    imgsz=640,
    device="cpu",
    )
```

```python
model.save(ROOT_DIR+"/exported.pt")
```

Figure 10: Code to Train & Export YOLO Model.

3.4 Model Validation

- Model is validated on unseen dataset.

Table 2: Class wise Performance of YOLO Model.

Class	Precision (P)	Recall (R)	mAP@50	mAP@50-95
figure_with_label	1.000	0.968	0.995	0.99
figure_without_label	0.997	1.00	0.995	0.985
graph	0.958	1.00	0.995	0.995

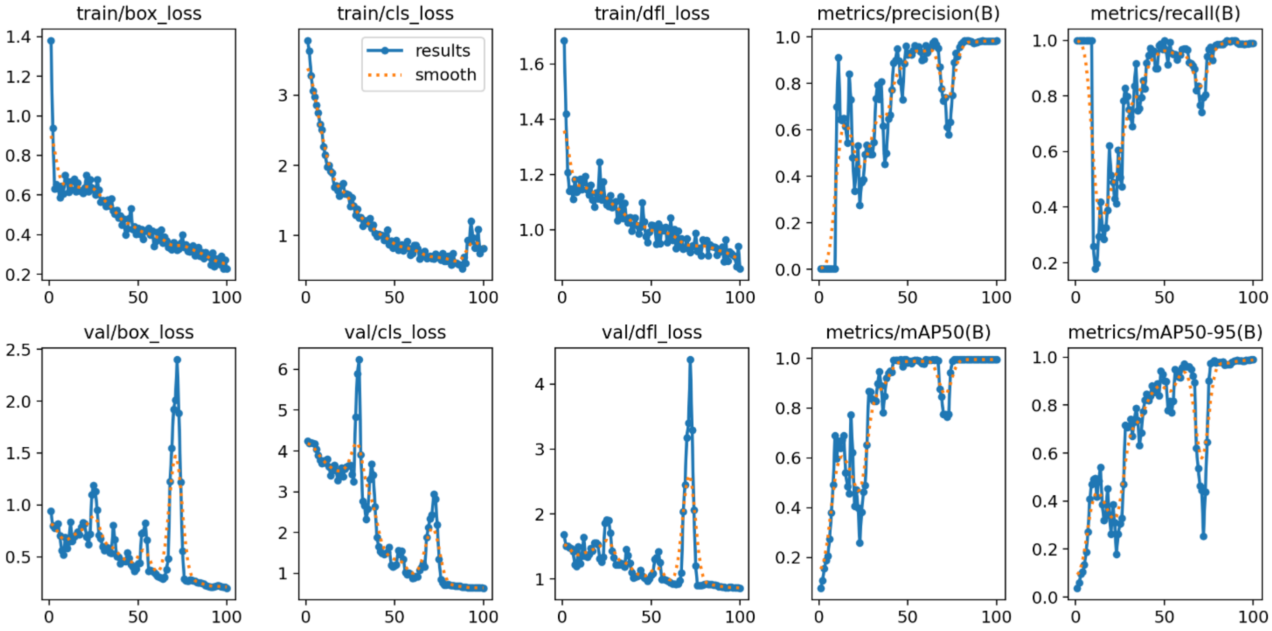


Figure 11: Training and Validation Metrics of YOLOv8 Model Over 100 Epochs.

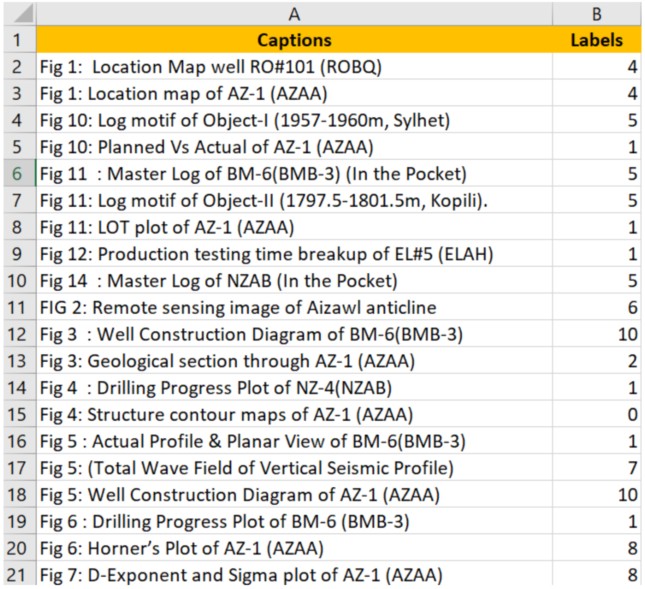
The trained YOLOv8 model achieved high accuracy in detecting images in well completion reports, with an mAP@50 of 0.995. The detection speed is optimal for real-time processing.

CHAPTER 4

TRAINING CAPTION CLASSIFIER (NLP)

4.1 Data Collection

- Captions were extracted from multiple well completion reports using PyMuPDF.
- The dataset contained labelled captions, stored in a CSV file.

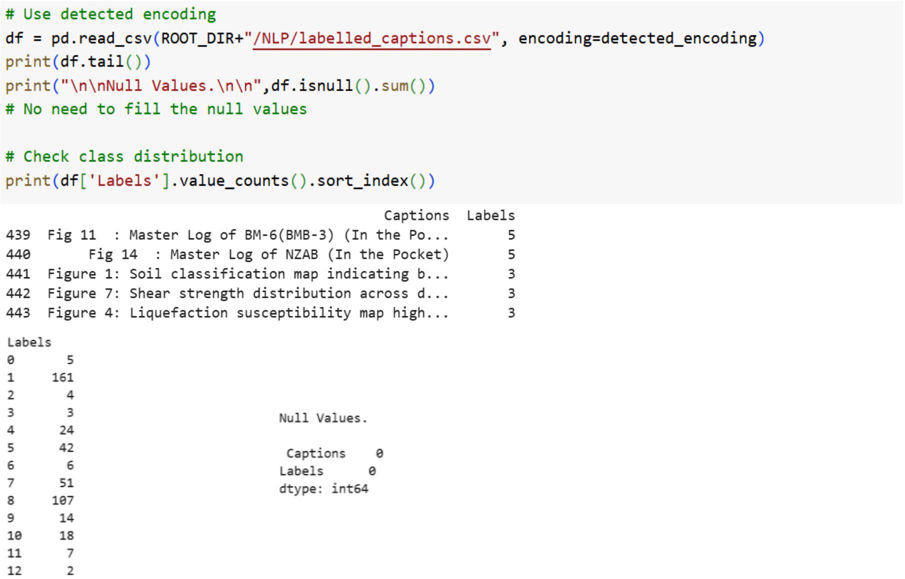


	A	B
1	Captions	Labels
2	Fig 1: Location Map well RO#101 (ROBQ)	4
3	Fig 1: Location map of AZ-1 (AZAA)	4
4	Fig 10: Log motif of Object-I (1957-1960m, Sylhet)	5
5	Fig 10: Planned Vs Actual of AZ-1 (AZAA)	1
6	Fig 11 : Master Log of BM-6(BMB-3) (In the Pocket)	5
7	Fig 11: Log motif of Object-II (1797.5-1801.5m, Kopili).	5
8	Fig 11: LOT plot of AZ-1 (AZAA)	1
9	Fig 12: Production testing time breakup of EL#5 (ELAH)	1
10	Fig 14 : Master Log of NZAB (In the Pocket)	5
11	FIG 2: Remote sensing image of Aizawl anticline	6
12	Fig 3 : Well Construction Diagram of BM-6(BMB-3)	10
13	Fig 3: Geological section through AZ-1 (AZAA)	2
14	Fig 4 : Drilling Progress Plot of NZ-4(NZAB)	1
15	Fig 4: Structure contour maps of AZ-1 (AZAA)	0
16	Fig 5 : Actual Profile & Planar View of BM-6(BMB-3)	1
17	Fig 5: (Total Wave Field of Vertical Seismic Profile)	7
18	Fig 5: Well Construction Diagram of AZ-1 (AZAA)	10
19	Fig 6 : Drilling Progress Plot of BM-6 (BMB-3)	1
20	Fig 6: Horner's Plot of AZ-1 (AZAA)	8
21	Fig 7: D-Exponent and Sigma plot of AZ-1 (AZAA)	8

Figure 12: Gathered and Manually Labelled Dataset.

4.2 Data Pre-processing

- To import the csv firstly detected its encoding than imported the whole csv (comma separated values) file.

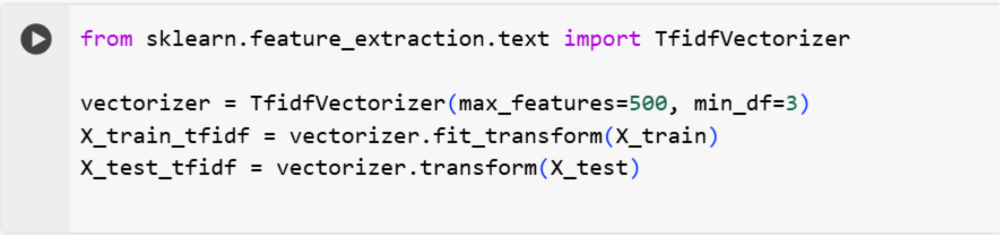


```
# Use detected encoding
df = pd.read_csv(ROOT_DIR+"/NLP/labelled_captions.csv", encoding=detected_encoding)
print(df.tail())
print("\n\nNull Values.\n\n",df.isnull().sum())
# No need to fill the null values

# Check class distribution
print(df['Labels'].value_counts().sort_index())
```

```
                                        Captions  Labels
439  Fig 11  : Master Log of BM-6(BMB-3) (In the Po...       5
440       Fig 14  : Master Log of NZAB (In the Pocket)       5
441  Figure 1: Soil classification map indicating b...       3
442  Figure 7: Shear strength distribution across d...       3
443  Figure 4: Liquefaction susceptibility map high...       3

Labels
0       5
1     161
2       4
3       3                          Null Values.
4      24
5      42                           Captions    0
6       6                          Labels      0
7      51                          dtype: int64
8     107
9      14
10     18
11      7
12      2
```

Figure 13: Checking the Labelled Captions Dataset.

- Checked for Null values and Class Distribution.
- The data was shuffled for randomness and split into 80% training and 20% testing.
- Text Processing –
    1. Lowercasing
    2. Punctuation removal
    3. Tokenization
    4. Stop word removal
    5. Lemmatization
- Feature Extraction using TF-IDF (Term Frequency-Inverse Document Frequency) to convert them to numeric values.



```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(max_features=500, min_df=3)
X_train_tfidf = vectorizer.fit_transform(X_train)
X_test_tfidf = vectorizer.transform(X_test)
```

Figure 14: Feature Extraction using TF-IDF Vectorization.

- SMOTE (Synthetic Minority Over-sampling Technique) was used to balance class distribution.

4.3 Model Training

- Five different machine learning models were trained:
    a. Logistic Regression
    b. Naïve Bayes
    c. Support Vector Machine (SVM)
    d. Random Forest
    e. XGBoost

  The best-performing model was identified based on accuracy metrics Logistic Regression.

Table 3: Models Accuracy Summary.

Model	Training Accuracy	Testing Accuracy
Logistic Regression	0.9976	0.9888
Naïve Bayes	0.9911	0.9213
SVM	0.9988	0.9888
Random Forest	0.9988	0.9888
XGBoost	0.9988	0.9663

4.4 Model Evaluation

The confusion matrix shows how well the model classified different caption types.

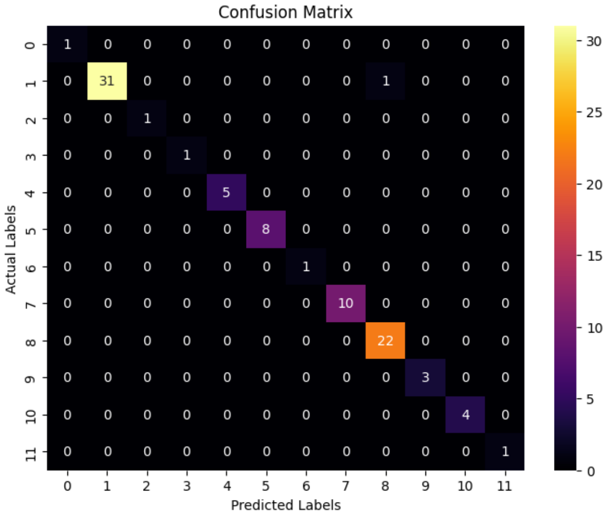


Figure 15: Confusion Matrix for Logistic Regression.

4.5 Model Saving

- The best-performing model (Logistic Regression) was saved for future use.
- The trained model and TF-IDF vectorizer were saved as:
    1. *logistic_regression_model.pkl*
    2. *tfidf_vectorizer.pkl*

CHAPTER 5

USING TRAINED MODELS TO SETUP THE FINAL SYSTEM

This chapter details the deployment of the trained YOLOv8 image detection model and Logistic Regression-based caption classifier to process well completion reports in PDF format. The system extracts images from PDFs using YOLOv8, captures captions using PyMuPDF (fitz), classifies images based on detected text, and organizes them accordingly. The entire workflow is designed for efficient extraction and classification.

5.1 Required Packages

1. Ultralytics
2. PyQt6
3. Scikit-learn
4. PyMuPDF
5. OpenCV_Python
6. Joblib
7. Pillow

5.2 System Workflow

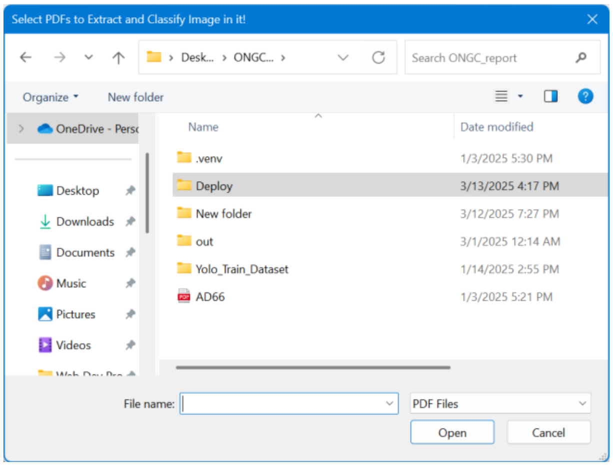
The complete system workflow involves:

- PDF Processing: Convert each page into an image.
- Object Detection: Detect figures, graphs, and labelled/unlabelled images using YOLO.
- Caption Extraction: Identify captions near detected images.
- Caption Classification: Use an NLP model to categorize images based on captions.
- Data Organization: Save classified images and captions into structured directories.
- GUI-based Interaction: Provide users with a file selection dialog and a success popup with easy access to outputs.

5.3 Browsing Pdfs

- The user is allowed to select one or more pdf file from an Interactive browse popup.
- User is restricted to select only pdf files.
- This captures the absolute addresses to those pdf files.

- It will keep on calling the *detect_and_classify()* function for all those addresses.



Select PDFs to Extract and Classify Image in it!

Name	Date modified
.venv	1/3/2025 5:30 PM
Deploy	3/13/2025 4:17 PM
New folder	3/12/2025 7:27 PM
out	3/1/2025 12:14 AM
Yolo_Train_Dataset	1/14/2025 2:55 PM
AD66	1/3/2025 5:21 PM

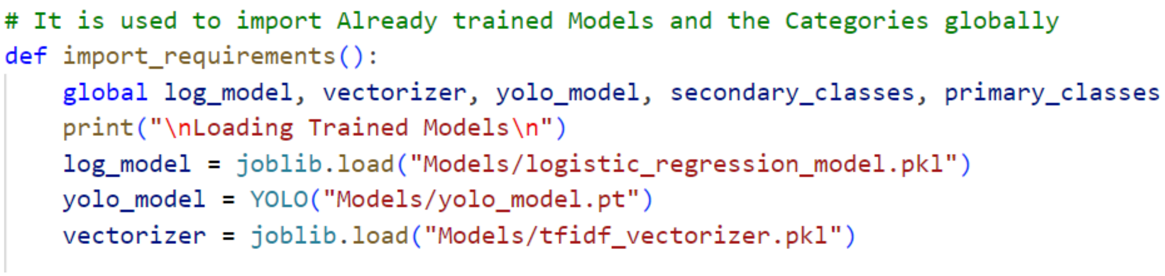
File name:

PDF Files

Figure 16: Pop to select one or more PDF files.

5.4 Importing the Required Models

- To ensure the system runs efficiently, the required pre-trained models and classification categories are imported globally. The YOLO model is loaded for image detection, while the Logistic Regression model and TF-IDF vectorizer are used for caption classification.
- This ensures the YOLO model is available for object detection, and the Logistic Regression model is ready for text classification.



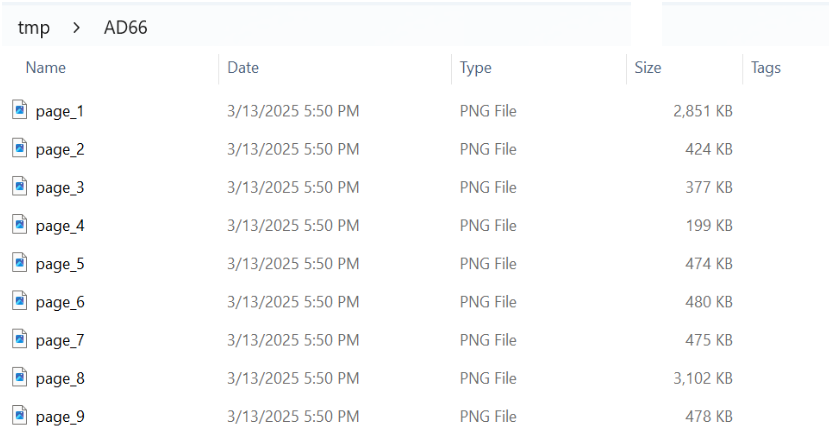
```python
# It is used to import Already trained Models and the Categories globally
def import_requirements():
    global log_model, vectorizer, yolo_model, secondary_classes, primary_classes
    print("\nLoading Trained Models\n")
    log_model = joblib.load("Models/logistic_regression_model.pkl")
    yolo_model = YOLO("Models/yolo_model.pt")
    vectorizer = joblib.load("Models/tfidf_vectorizer.pkl")
```

Figure 17: Importing Saved Model for Prediction.

5.5 Converting PDFs to Images

- To process the PDF, each page is converted into a high-resolution image (300 dpi) using PyMuPDF.

- The images are exported to '*tmp*' folder in current directory, once the detection is completed this folder with all its image is deleted.



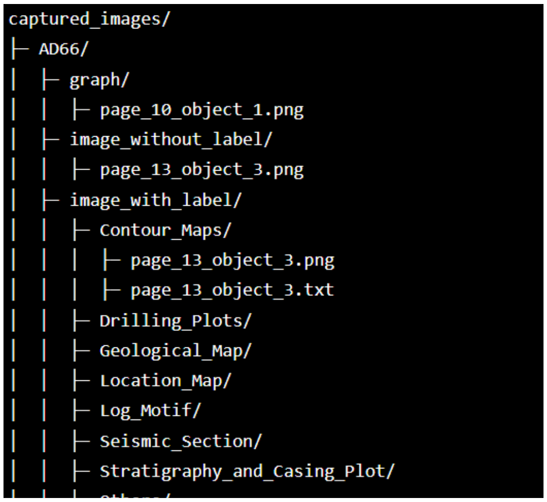
tmp > AD66

Name	Date	Type	Size	Tags
page_1	3/13/2025 5:50 PM	PNG File	2,851 KB	
page_2	3/13/2025 5:50 PM	PNG File	424 KB	
page_3	3/13/2025 5:50 PM	PNG File	377 KB	
page_4	3/13/2025 5:50 PM	PNG File	199 KB	
page_5	3/13/2025 5:50 PM	PNG File	474 KB	
page_6	3/13/2025 5:50 PM	PNG File	480 KB	
page_7	3/13/2025 5:50 PM	PNG File	475 KB	
page_8	3/13/2025 5:50 PM	PNG File	3,102 KB	
page_9	3/13/2025 5:50 PM	PNG File	478 KB	

Figure 18: Images of Exported Page.

5.6 Object Detection Using YOLO

- Once pages are extracted as images, the YOLO model processes each page to detect images and graphs & the detected objects are cropped and saved in directories of the class they belong.



```
captured_images/
├ AD66/
|  ├ graph/
|  |  ├ page_10_object_1.png
|  ├ image_without_label/
|  |  ├ page_13_object_3.png
|  ├ image_with_label/
|  |  ├ Contour_Maps/
|  |  |  ├ page_13_object_3.png
|  |  |  ├ page_13_object_3.txt
|  |  ├ Drilling_Plots/
|  |  ├ Geological_Map/
|  |  ├ Location_Map/
|  |  ├ Log_Motif/
|  |  ├ Seismic_Section/
|  |  ├ Stratigraphy_and_Casing_Plot/
|  |  ├ Others/
```

Figure 19: Output directory structure.

- At every detection of image on a page, the Captions is searched in the full width of page and for the height of image + 50 below it.
- The caption is searched as it starts from "*fig*" (case insensitive) and it keeps capturing to next line if it ends with "-".

- The caption is saved in text file in the same folder to image with the same file name.

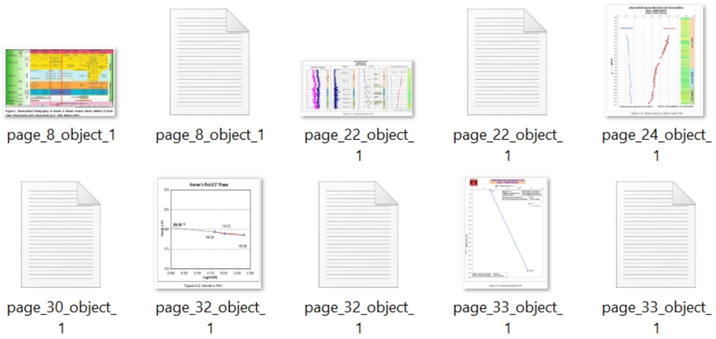
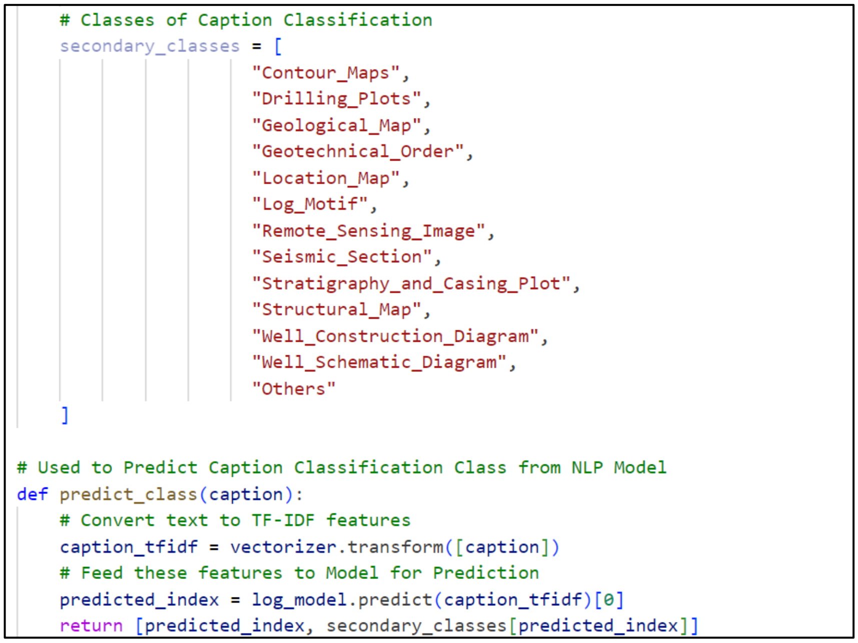


Figure 20: Exported Images and their Captions.

5.7 Caption Classification Using NLP

- The caption captured in previous stage is simultaneously vectorized and then fed to the Logistic Regression which gives out the index label.



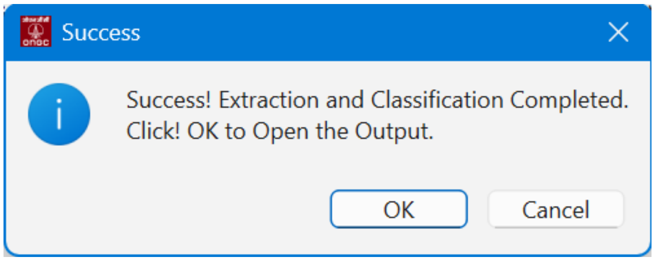
```
    # Classes of Caption Classification
    secondary_classes = [
                    "Contour_Maps",
                    "Drilling_Plots",
                    "Geological_Map",
                    "Geotechnical_Order",
                    "Location_Map",
                    "Log_Motif",
                    "Remote_Sensing_Image",
                    "Seismic_Section",
                    "Stratigraphy_and_Casing_Plot",
                    "Structural_Map",
                    "Well_Construction_Diagram",
                    "Well_Schematic_Diagram",
                    "Others"
    ]

# Used to Predict Caption Classification Class from NLP Model
def predict_class(caption):
    # Convert text to TF-IDF features
    caption_tfidf = vectorizer.transform([caption])
    # Feed these features to Model for Prediction
    predicted_index = log_model.predict(caption_tfidf)[0]
    return [predicted_index, secondary_classes[predicted_index]]
```

Figure 21: Classes for Caption Classification (Secondary Classes).

- This label is used to find the class which it belongs and hence the folder it will be saved with the caption.

5.8 Success Popup

- At final completion of classification, a popup appears saying “Success, Click OK to open Output Folder”.
- When clicked on OK it open Output folder.



Success! Extraction and Classification Completed.
Click! OK to Open the Output.

Figure 22: Success Popup triggered to open output folder.

CHAPTER 6

CONCLUSION & FUTURE WORK

6.1 Conclusion

This project successfully implemented an automated system for extracting and classifying images from well completion report PDFs using a combination of computer vision and natural language processing techniques. The trained YOLOv8 model accurately detected figures, graphs, and labelled images within the PDF pages, while the Logistic Regression-based NLP model effectively classified extracted captions. The system efficiently processes multiple PDFs at once, organizes extracted content into structured directories, and provides a user-friendly interface via PyQt6 for streamlined interaction.

The results demonstrate that using deep learning for object detection and machine learning for text classification can significantly enhance the efficiency of processing complex technical reports. The integration of PyMuPDF for caption extraction further improved the automation level, minimizing manual intervention. Additionally, the graphical user interface ensures ease of access for users, making the system practical and effective.

6.2 Future Work

Although the system performs well in extracting and classifying images and captions, there are potential improvements that can be explored in future work. Some key enhancements include:

- ✓ Improving Caption Extraction: Enhancing text extraction to better handle multi-line captions, rotated text, and images with embedded descriptions.
- ✓ Deep Learning-based Caption Classification: Experimenting with transformer-based models such as BERT or GPT to improve the classification accuracy of captions.
- ✓ Extending Image Classification: Implementing CNN-based models to further categorize detected images into more detailed subcategories.
- ✓ Handling OCR for Non-Textual Captions: Integrating Optical Character Recognition (OCR) techniques to extract information from embedded text inside images.
- ✓ Enhancing GUI Features: Adding progress indicators, real-time feedback, and batch processing capabilities in the PyQt6 interface.

- ✓ Cloud-Based Deployment: Extending the project by deploying the system as a web application with cloud storage for enhanced accessibility and scalability.
- ✓ Performance Optimization: Reducing inference time by optimizing model parameters and leveraging GPU acceleration for faster processing.
- ✓ Multilingual Support: Expanding the NLP model to support caption classification in multiple languages.

By incorporating these improvements, the system can become even more robust and versatile for analysing complex technical reports. Future advancements in AI and NLP techniques will further refine the automation and accuracy of such document processing applications.