IT255 - Multimodal Complaint Registration System

Anush Revankar - 221AI009
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: anushrevankar.221ai009@nitk.edu.in

Akhilesh Negi - 221AI008
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: aknegi.221ai008@nitk.edu.in

Adithya V -221AI005
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: adithyav.221ai005@nitk.edu.in

Abhaysingh - 221AI002
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: abhayrajput.221ai002@nitk.edu.in

***Abstract*—In today's highly connected society, there is a significant demand for efficient complaint resolution especially for muncipal corporations. Our project aims to address this need by utilizing technology to develop a versatile system that caters to users' communication preferences. This system allows users to raise issues through voice, text, or images. By incorporating advanced techniques such as LSTM, Convolutional Neural Networks, and YOLOv8, we strive to create a strong multimodal complaint management system specifically designed for municipal corporations. Through the integration of voice-to-text conversion and image captioning, our project enhances accessibility and improves the effectiveness of resolving complaints.**

Keywords: Complain Registration System, AI applications, NLP

I. Introduction

Efficient and smooth complaint resolution is of utmost importance in today's interconnected world, as it directly impacts user/customer satisfaction and organizational integrity. This holds especially true for municipal corporations. Given the increasing dependency on technology, it is necessary to create a complaint system that caters to the various communication preferences within municipal settings. The main obstacle we face is the inefficiencies and accessibility limitations in traditional complaint resolution systems that are designed for municipal corporations.

Our idea allows users to file complaints using different channels (voice, text, or images) and receive instant updates on their progress. This comprehensive method promotes transparency and accessibility during the resolution process, which is essential for municipal corporations to maintain their dedication to public service.

The importance of this project is rooted in its direct influence on user satisfaction and the trustworthiness of local government entities. Insufficient resolving of complaints can result in dissatisfied citizens and damage the reputation of municipal administrations. Our system improves user experience and maintains the integrity of municipal corporations by simplifying the process of submitting complaints and providing timely updates. This problem is complex due to the need to seamlessly integrate different technologies to cater to diverse communication preferences, while also guaranteeing the accuracy and relevancy of the complaints received. Already existing solutions often fall short in addressing the distinct challenges faced by municipal corporations, highlighting the need for a sophisticated approach.

The central idea of our project is to utilize AI technologies, such as natural language processing (NLP) and computer vision, to process complaints submitted via voice, text, or images. Through techniques such as voice-to-text conversion, image caption generation, and AI-based topic constraint satisfaction problem (CSP) for complaint validation, we aim to create a robust and efficient complaint resolution system. Driven by our commitment to enhancing user experience and organizational efficiency, our project addresses the shortcomings of existing complaint resolution systems.

II. Literature Survey

The Multimodal Complaint Registration system was developed using AI and ML techniques. YOLOv8 was fine-tuned for object detection and image classification, while LSTM models were employed for image captioning and text classification. Additionally, a CNN model was trained from scratch on a custom dataset for image classification. The system was integrated into a Django-based website for user interaction.

In recent research, significant advancements have been made in the field of multimodal complaint analysis and classification. For instance, Ignatious et al. (2019) proposed a Semantic Driven CNN–LSTM Architecture for Personalised Image Caption Generation, which focuses on generating personalized captions for images using semantic information extracted from CNN features and LSTM models [1]. Chen and Qin (2022) explored the use of SinaWeibo microblogs for identifying food customer complaints, employing natural language processing techniques to analyze customer feedback [2]. Shi et al. (2023) investigated the recognition and classification of recyclable garbage using an improved YOLOv8s model,

TABLE I: Summary of Literature Survey

Authors	Methodology	Remarks
L. Abisha Anto Ignatious., S. Jeevitha., M. Madhurambigai., and M. Hemalatha (2019)	The authors propose a semantic-driven architecture that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. They extract semantic information from CNN features and incorporate it into LSTM-based caption generation. The model aims to generate personalized image captions by considering both visual features and semantic context.	The paper addresses the challenging task of personalized image captioning, which requires understanding visual content and context. By leveraging both CNNs and LSTMs, the proposed architecture captures both visual and semantic information effectively. This methodology could be beneficial in applications where personalized image descriptions are required, such as image recommendation systems or assistive technologies for visually impaired individuals.
F. Chen and W. Qin (2022)	The authors utilize SinaWeibo microblogs, a social media platform, to identify complaints related to food customers. They employ natural language processing techniques to analyze textual data from microblogs, focusing on identifying complaints and sentiment analysis. Machine learning algorithms are applied to classify microblogs into complaint categories and assess customer sentiment.	This study demonstrates the potential of leveraging social media data for customer complaint analysis, offering insights into customer opinions and preferences. By using machine learning techniques for sentiment analysis and complaint categorization, the methodology provides an automated approach to handle large volumes of customer feedback efficiently. The findings could assist food businesses in addressing customer concerns and improving overall satisfaction.
Y. Shi, X. Li, G. Wang, and X. Jin, (2023)	The authors propose an improved version of the YOLOv8 model for the recognition and classification of recyclable garbage. They enhance the robustness of YOLOv8 in complex environments by incorporating attention mechanisms and deep learning techniques. The model focuses on accurately detecting and classifying recyclable objects in diverse environmental conditions.	This research addresses the critical issue of waste management by developing a robust model for garbage recognition and classification. The incorporation of attention mechanisms and deep learning enhances the model's ability to handle variations in environmental factors, such as lighting conditions and object orientation. The methodology has potential applications in smart waste management systems, contributing to environmental sustainability efforts.
M. Gupta, A. Singh, R. Jain, A. Saxena, and S. Ahmed (2021)	The authors propose RailNeural, an Attention-Based Bi-Directional LSTM model for multi-class railway complaints categorization. They utilize an official dataset from the COMS app to train the model, capturing character-level features of user complaint input sequences. The model classifies complaints into their respective departments, enabling prompt and accurate redressal of railway-related issues.	RailNeural addresses the challenge of efficiently categorizing and addressing railway-related complaints, which are crucial for passenger satisfaction and efficient operations. By leveraging deep learning techniques, including LSTM networks and attention mechanisms, the model achieves high accuracy in classifying complaints. The methodology offers a scalable and effective approach to handling large volumes of railway complaints, potentially reducing response times and improving service quality.
Y. Zhang, (2021)	The author investigates text classification methods based on Long Short-Term Memory (LSTM) neural network models. The study focuses on training LSTM models for text classification tasks, such as sentiment analysis and categorization. Techniques such as word embedding and sequential processing are utilized to capture textual features and dependencies.	This research contributes to the field of natural language processing by exploring LSTM-based methods for text classification. By leveraging the sequential nature of text data, LSTM models can effectively capture contextual information and dependencies, leading to improved classification performance. The methodology offers a versatile approach to text classification tasks, with potential applications in sentiment analysis, document categorization, and more.

enhancing the robustness of object detection in complex environments [3]. Gupta et al. (2021) developed RailNeural, an Attention-Based Bi-Directional LSTM model for multi-class railway complaints categorization, achieving high accuracy and prompt complaint redressal [4]. Zhang (2021) proposed a text classification method based on LSTM neural network models, focusing on sentiment analysis and text categorization [5].

In comparison to these recent works, our Multimodal Complaint Registration system offers a comprehensive solution by integrating multiple AI and ML models for object detection, image captioning, and text classification. While Ignatious et al. (2019) and Zhang (2021) focus on specific aspects such as image captioning and text classification, our system provides a holistic approach to complaint registration by addressing both image and text inputs. Additionally, our use of YOLOv8 for object detection and CNN models for image classification contributes to the system's robustness in handling diverse complaints.

III. Problem Statement

Development of a sophisticated complaint registration system which will utilize multimodal AI capabilities. This system should allow users to submit complaints using voice, text, or images, and provide real-time updates on the status of the complaint through voice and text messages. The objective is to enhance accessibility and efficiency in resolving complaints by utilizing advanced AI models that can handle various complaint formats.

A. Objectives

1) **Implement Multimodal AI Integration:** Develop a robust framework integrating various AI models capable of processing voice, text, and image inputs for complaint registration. Use advanced techniques such as natural language processing (NLP) and computer vision(YOlO,CNNs) to analyze and interpret complaints across multiple modalities.
2) **Provide smooth seamless navigation and usage of the complaint registration website:** Develop a seamless complaint submission experience by developing a user-friendly interface that enables users to effortlessly submit complaints using voice, text, or image inputs. Incorporate intuitive features that will guide users through the complaint submission process, ensuring it is easy to use and accessible for all.
3) **Implement Real-Time Status Updates**: Establish mechanisms for providing real-time updates on complaint status through voice and text messages, ensuring users are kept informed throughout the resolution process.
4) **Integrate Notification System:** Incorporate a notification system to deliver real-time updates to users regarding the progress of their complaints.

IV. Course Mappings

In our complaint registration system, we model the problem as a Constraint Satisfaction Problem (CSP), where:

Variables (C): Each complaint in our system is represented as a variable (C), uniquely identified and containing text along with additional metadata.

Domains (l_n): The domain (l_n) represents the length of the complaint text. We enforce a constraint specifying that the complaint must consist of a minimum of 40 characters, ensuring that registered complaints provide sufficient information for analysis and resolution.

Constraints (k_W): Constraints are imposed on the content of the complaint text to avoid spam or irrelevant complaints. These constraints involve specific keywords or phrases that must be present in the complaint text to be considered valid. Additionally, we enforce constraints such as content relevance, language and tone, duplicate complaint prevention, and priority thresholds to ensure the quality and appropriateness of registered complaints.

Heuristics are essential in efficiently and effectively guiding the search process. In our system, we utilize the following heuristics to organize and prioritize complaints:

a. Department Classification: Heuristic: Complaints are assigned to departments based on analyzing the content of the complaint text. This heuristic involves examining keywords or phrases within the complaint text to determine the relevant department for handling the complaint. For instance, if a complaint contains keywords like "water," it may be classified under the Water Department, indicating a higher priority due to the importance of water-related issues.

b. Priority Ranking: Heuristic: Complaints are prioritized by considering various factors, including the duration of unserviced complaints, the frequency of keyword matches, severity assessment, user reputation, resource availability, and escalation mechanisms. This heuristic aims to intelligently prioritize complaints rather than relying solely on a first-come, first-served approach. By taking into account factors such as severity, user reputation, and resource availability, we ensure that critical issues receive prompt attention while optimizing resource allocation and enhancing user satisfaction.

To ensure the quality and relevance of registered complaints, we enforce a minimum length constraint of 40 characters. This means that any complaint below this threshold will be filtered out. By doing so, we can guarantee that the complaints we receive provide enough information for thorough analysis and resolution.

Incorporating these variables and heuristics into our complaint registration system enables us to effectively manage and prioritize complaints, ensuring timely resolution.

Department Classification:

$$Priority_{\text{department}} = \sum_{i=1}^{n} \text{KeywordMatch}(c_i) \times \text{Weight}(k_i)$$

This formula calculates the priority of a complaint for a specific department based on the sum of keyword matches (weighted by importance) in the complaint text.

Priority Ranking:

$$Priority_{\text{complaint}} = \text{DaysUnserviced}(c) + \text{MaxKeywordMatches}(c)$$

This formula combines the number of days a complaint has remained unserviced with the maximum number of keyword matches in the complaint text to determine its priority for resolution.

Length Constraint Enforcement:

$$ValidComplaint = \begin{cases} 1 & \text{if Length}(c) \geq \text{MinLengthThreshold} \\ 0 & \text{otherwise} \end{cases}$$

This formula checks if a complaint meets the minimum length threshold, returning 1 if the length is sufficient and 0 otherwise. It ensures that only valid complaints are considered for further processing.

V. Methodology

The methodology consists of the following steps:

A. Data Collection and Preprocessing

1) Collect images and corresponding textual data (complaint descriptions).
2) Preprocess images (resize, normalization, augmentation).
3) Tokenize and preprocess textual data (remove stopwords, tokenize, pad sequences).

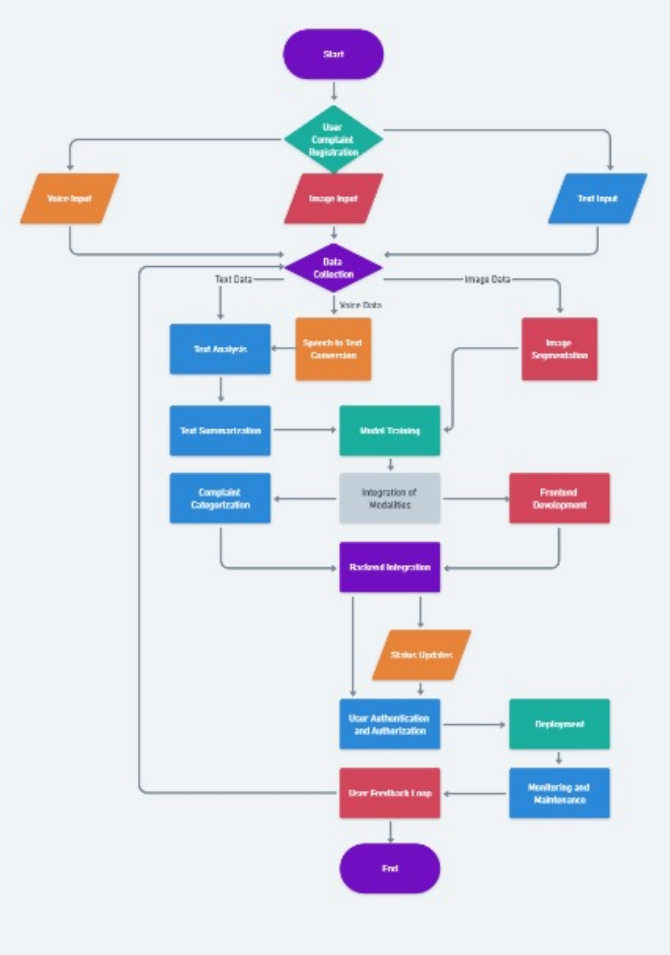

Fig. 1: Flow Diagram

4) Preprocess by removing stopwords, experimented with stemming and lemmatization and Word embedding techniques such as BOW and Word2Vec.
5) Also handling imbalanced dataset by Data Augmentation .
6) Voice Data: Use Google Speech-to-Text API to transcribe audio to text and multilingual input.
7) Image Data: Applied Image Augmentation, Resized the images and Normalized the image matrices. Finetuned various pretrained CNN model to extract image features.

B. Model Training

1) Split the dataset into training, validation, and testing sets.
2) Train YOLOv8 for object detection and image classification.
3) Train LSTM model for image captioning using the image-text pairs.
4) Train CNN model from scratch for image classification.
5) Fine-tune VGG16 model for image classification.
6) Compare all the image classification models. (YOLOv8 finetuned, CNN, VGG16).

C. Model Integration

1) Develop a Django-based website for the complaint registration system.
2) Integrate all models into the website backend.

D. Inference

1) Take user input (image and textual description) through the website.
2) Use YOLOv8 for object detection and image classification.
3) Generate captions using the LSTM model.
4) Perform text classification using LSTM model.
5) Display results on the website interface.

VI. Mathematical Functions and Algorithms

A. YOLOv8

Algorithm 1 YOLOv8-Inspired Classification Model

Input: Image of size (W, H)
Preprocess the image (resize, normalize, etc.)
Define grid size: $S \times S$ (e.g., $S = 7$)
Define number of classes: C (e.g., $C = 10$ for 10 classes)
Construct a convolutional neural network (CNN) architecture:
    Input layer: $(W, H, 3)$ image
    Convolutional layers followed by max pooling
    Fully connected layers for classification (no bounding box prediction)
Forward Pass:
    Pass the preprocessed image through the CNN
    Output: $(S \times S \times C)$ tensor
    Each grid cell predicts class probabilities for each of the C classes
Post-processing:
    Apply softmax activation to convert class scores into probabilities
    Extract the class with the highest probability as the predicted class for each grid cell
Output:
    Display or use the predicted class probabilities for the entire image or individual grid cells

1) Loss Function: The loss function in YOLOv8 consists of two components:

- Localization Loss:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{b}_i - b_i)^2$$

- Classification Loss:

$$\text{CrossEntropyLoss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij})$$

2) Intersection over Union (IoU):

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}}$$

B. LSTM Model

1) Loss Function (Text Classification):

$$\text{Categorical Cross-Entropy Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij})$$

2) Loss Function (Caption Generation):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

C. CNN Model (From Scratch)

1) Loss Function:

$$\text{Categorical Cross-Entropy Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij})$$

D. VGG16 (Fine-tuned)

1) Loss Function: Same as the CNN model from scratch, categorical cross-entropy loss is used.

E. Image Captioning Model Architecture

For image captioning, we use a model called Bidirectional LSTMs. Our model builds on the LSTM cell, which is a particular form of traditional recurrent neural network (RNN).

The reading and writing memory cell c is controlled by a group of sigmoid gates. At a given time step t, the LSTM receives inputs from different sources: current input x_t, the previous hidden state of all LSTM units h_{t-1}, as well as the previous memory cell state c_{t-1}. The updating of those gates at time step t for given inputs x_t, h_{t-1}, and c_{t-1} is as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{1}$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{2}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{3}$$
$$g_t = \phi(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{4}$$
$$c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \tag{5}$$
$$h_t = o_t \cdot \phi(c_t) \tag{6}$$

where W are the weight matrices learned from the network and b are bias vectors. σ is the sigmoid activation function $\sigma(x) = 1/(1+\exp(-x))$ and ϕ represents hyperbolic tangent $\phi(x) = (\exp(x) - \exp(-x))/(\exp(x) + \exp(-x))$. $\cdot$ denotes the products with a gate value. The LSTM hidden output $h_t = \{h_{tk}\}_{k=0}^{K}$, $h_t \in R^K$ will be used to predict the next word by the Softmax function with parameters W_s and b_s:

$$F(p_{ti}; W_s, b_s) = \frac{\exp(W_s h_{ti} + b_s)}{\sum_{j=1}^{K}\exp(W_s h_{tj} + b_s)} \tag{7}$$

where p_{ti} is the probability distribution for the predicted word. Our key motivation for choosing LSTM is that it can learn long-term temporal activities and avoid the quick exploding and vanishing problems that traditional RNN suffers from during backpropagation optimization.

The bidirectional LSTM is implemented with two separate LSTM layers for computing forward hidden sequences $\rightarrow h$ and backward hidden sequences $\leftarrow h$. The forward LSTM starts at time $t = 1$ and the backward LSTM starts at time $t = T$. Formally, our model works as follows: for raw image input I_e, forward order sentence $\rightarrow S$ and backward order sentence $\leftarrow S$, the encoding performs as:

$$I_t = C(I_e; \Theta_v) \tag{8}$$
$$\rightarrow h_1^T = T(\rightarrow E \rightarrow S; \rightarrow \Theta_l) \tag{9}$$
$$\leftarrow h_1^T = T(\leftarrow E \leftarrow S; \leftarrow \Theta_l) \tag{10}$$

where C, T represent CNN, T-LSTM respectively and Θ_v, Θ_l are their corresponding weights. $\rightarrow E$ and $\leftarrow E$ are bidirectional embedding matrices learned from the network. Encoded visual and textual representations are then embedded to multimodal LSTM by:

$$\rightarrow h_2^T = M(\rightarrow h_1^T, I_t; \rightarrow \Theta_m) \tag{11}$$
$$\leftarrow h_2^T = M(\leftarrow h_1^T, I_t; \leftarrow \Theta_m) \tag{12}$$

where M presents M-LSTM and its weight Θ_m. M aims to capture the correlation of visual context and words at different time steps. We feed the visual vector I to the model at each time step for capturing strong visual-word correlation. On top of M-LSTM are Softmax layers which compute the probability distribution of the next predicted word by:

$$\rightarrow p_{t+1} = F(\rightarrow h_2^T; W_s, b_s) \tag{13}$$
$$\leftarrow p_{t+1} = F(\leftarrow h_2^T; W_s, b_s) \tag{14}$$

where $p \in R^K$ and K is the vocabulary size.

Initially, we have a set of images and a set of captions for each of the images. Then we extract image features for each image using a pretrained VGG16 model by removing the last two layers. Then we preprocess the captions by removing stopwords, applying lemmatization, and then create a dictionary of image names and their captions.

We also used Data Augmentation for creating more images as each set of problems had just 200 images. Data Augmentation included random cropping, horizontal flip, vertical flip, etc.

Then we pass both feature dictionaries and image caption dictionaries to the LSTM model for training.

F. Training of the Image Captioning Model

Our model is trained using Stochastic Gradient Descent (SGD). The joint loss function $L = \rightarrow L + \leftarrow L$ is computed by accumulating the Softmax losses of forward and backward directions. Our main focus is to minimize L, which is the same as maximizing the probabilities of correctly generated sentences. We calculate the gradient ∇L with Back-Propagation Through Time (BPTT) algorithm. The trained model is used to predict a word w_t with a given image context I and previous word context $w_{1:t-1}$ by $P(w_t|w_{1:t-1}, I)$ in forward order, or

by $P(w_t|w_{t+1:T}, I)$ in backward order. We set $w_1 = w_T = 0$ at the start point respectively for forward and backward directions. Ultimately, with generated sentences from two directions, we decide the final sentence for a given image $p(w_{1:T}|I)$ according to the summation of word probability within the sentence.

G. Text Classification Model

Naive Bayes is a probabilistic classifier based on Bayes' theorem with the assumption of independence between features. The classifier predicts the probability of a class given the features by maximizing the posterior probability, as per Bayes' theorem:

$$P(Y|X) = \frac{P(X|Y) \times P(Y)}{P(X)}$$

. With the Naive Bayes assumption, the likelihood $P(X|Y)$ is simplified to the product of individual feature probabilities:

$$P(X|Y) = P(x_1|Y) \times P(x_2|Y) \times ... \times P(x_n|Y)$$

. Hence, the classifier predicts the class $\hat{y}$ that maximizes the expression:

$$\hat{y} = \underset{y}{\text{argmax}}\ P(Y = y) \times \prod_{i=1}^{n} P(x_i|Y = y)$$

. Parameter estimation involves calculating the prior probabilities $P(Y)$ and the conditional probabilities $P(x_i|Y)$ from the training data, typically through counting occurrences. Smoothing techniques may be applied to handle zero probabilities. In essence, Naive Bayes offers a simple yet effective method for classification by leveraging probabilities and feature independence.

VII. Experimental Results and Analysis

TABLE II: Text Models by ML Approach

Model	Accuracy	F1 Score
Naïve Bayes	0.9255	0.881
Word2vec + Logistic Regression	0.8238	0.82
Linear Support Vector Machine	0.8166	0.873

TABLE III: Text Models by DL Approach

Model	Accuracy
BOW + Keras	0.792
Word2vec + LSTM	0.8122

TABLE IV: Image Models

Model	Accuracy
Fine-tuned YOLO V8	0.91
Fine-tuned VGG16	0.86
CNN for Image Classification	0.70

Algorithm 2 LSTM-based Text Classification

Model Building:
- Initialize LSTM-based neural network architecture.
- Include an embedding layer to convert words into dense vectors.
- Add LSTM layers to capture sequential information.
- Utilize additional layers like pooling and dense layers for classification.
- Configure the output layer with softmax activation for multiclass classification.

Model Training:
- Split the dataset into training, validation, and test sets.
- Train the LSTM model on the training data.
- Tune hyperparameters such as learning rate and batch size for optimal performance.
- Validate the model on the validation set.

Model Evaluation:
- Evaluate the trained model's performance using metrics like accuracy, precision.
- Assess the model's ability to classify text samples into multiple classes accurately.

Inference:
- Deploy the trained model for making predictions on new, unseen text data.
- Preprocess incoming text data using the same preprocessing steps applied during training.
- Utilize the model to classify text samples into respective classes.

Monitoring and Optimization:
- Fine-tune the model and adjust hyperparameters as necessary.
- Monitor the model's performance to maintain or improve performance.

A. YOLOv8:

1) Experimental Setup:

a) Dataset: Custom dataset containing four classes:

- Road Problem
- Water Problem
- Electricity Problem
- Waste Problem

with approximately 200 images per class.

b) Model: YOLOv8.1.24 architecture was utilized for fine-tuning on the custom dataset.

c) Training Details:

- Epochs: 10
- Batch size: 16
- Image size: 64x64
- Optimizer: AdamW with automatic learning rate and momentum determination.

2) Parameters and Metrics:

a) Parameters:

- Epochs: 10
- Batch size: 16
- Image size: 64x64
- Optimizer: AdamW with automatic learning rate and momentum determination.

b) Metrics:

- Loss: Calculated during training to monitor convergence.
- Accuracy: Top-1 accuracy measured on the validation set.
- Inference Speed: Preprocess, inference, loss, and postprocess times per image.

3) Observations:

a) Training Progress:

- Loss decreased steadily over epochs, indicating effective learning.
- Achieved a top-1 accuracy of 91% on the validation set by the end of training.

b) Inference Speed:

- Inference speed per image:
  - Preprocess: 0.0ms
  - Inference: 0.6ms
  - Loss: 0.0ms
  - Postprocess: 0.0ms

c) Comparison with Existing Works: No direct comparison with existing works was provided in the provided information.

4) Analysis:

- The YOLOv8 model achieved promising results after fine-tuning on the custom dataset, with a validation accuracy of 91%.
- Inference speed was efficient, with low processing times per image, making it suitable for real-time applications.
- The provided setup and parameters seem effective for the given task, yielding satisfactory results.

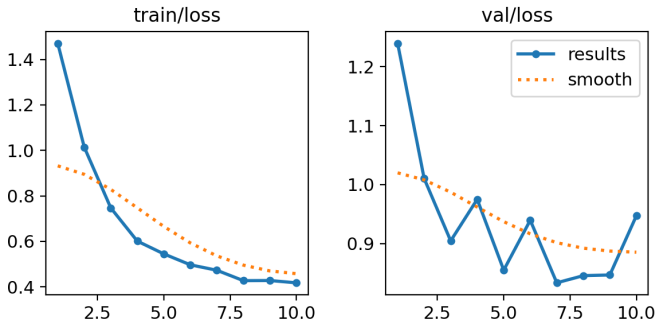

Fig. 2: YOLOv8: Validation Loss vs Training Loss

VIII. Conclusion

In conclusion, the development of the Multimodal Complaint Registration System marks a significant advancement in leveraging Artificial Intelligence (AI) and Machine Learning (ML) techniques to address real-world issues efficiently.

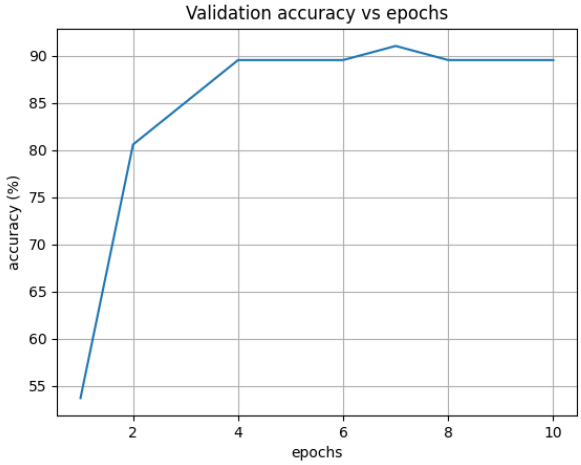

Fig. 3: YOLOv8: Validation Accuracy vs Epoch in YOLOv8. The model generated after 7th epoch (Accuracy: 91%) is integrated in the project.

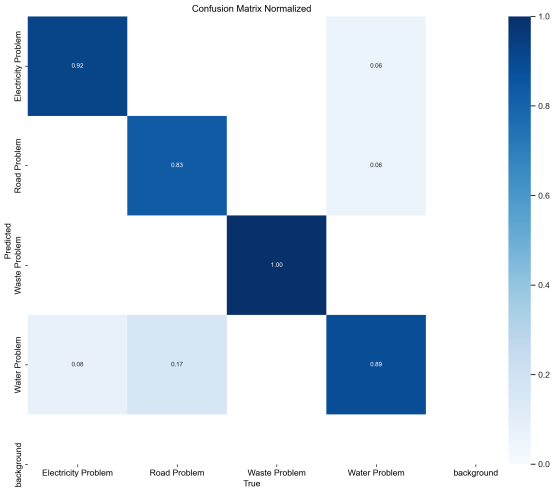

Fig. 4: YOLOv8: Normalized Confusion Matrix generated after the training. As you can see, most of the tests lead to correct results.

Through the integration of various AI models including YOLOv8 for object detection, LSTM for image captioning and text classification, CNN for image classification, and VGG16 for fine-tuning, we have created a comprehensive system capable of handling multimodal inputs for complaint registration.

The methodology involved meticulous steps, starting from data collection and preprocessing to model training, integration, and inference. By fine-tuning YOLOv8 with an impressive accuracy of 91%, we ensured robust object detection and image classification capabilities, which are crucial for identifying and categorizing complaints accurately. Additionally, the utilization of LSTM for generating captions and performing text classification adds a layer of context understanding to the

system, enhancing its overall performance.

The development of a Django-based website serves as an intuitive interface for users to interact with the system seamlessly, allowing them to register complaints by providing both images and textual descriptions. The integration of all models into the website backend streamlines the entire process, providing users with prompt and accurate feedback on their submitted complaints.

Overall, the Multimodal Complaint Registration System demonstrates the potential of AI and ML in addressing societal challenges effectively. Moving forward, continual refinement and optimization of the system can further enhance its performance and expand its applicability to various domains beyond complaint registration, ultimately contributing to the improvement of community welfare and resource allocation.

A. *User Interface*

The Complaint Registration System is a comprehensive web application designed to facilitate user complaint management efficiently. Upon logging in, users are presented with a dynamic dashboard providing an overview of various complaint categories, including Resolved, Pending, and Rejected Complaints, along with an option to view All Complaints. Additionally, users can conveniently access a list of their registered complaints, each displaying its current status.

User Dashboard Features:

- Resolved Complaints: Displays the number of complaints that have been successfully resolved.
- Pending Complaints: Shows the count of complaints awaiting resolution.
- Rejected Complaints: Indicates the number of complaints that were rejected.
- All Complaints: Offers a comprehensive view of all registered complaints.
- Lodge a New Complaint: Provides users with the ability to submit new complaints in various formats, including text, voice, or image.

Complaint Submission Process:

- Upon selecting the ”Lodge a new complaint” option, users are directed to the complaint submission page.
- Users can choose from multiple complaint formats, including text, voice, or image, based on their preference.
- The submission process is intuitive and user-friendly, allowing users to easily provide necessary details for their complaint.

Staff Dashboard Features:

- Staff members have access to a dedicated dashboard that categorizes complaints based on departmental divisions.
- Complaints are automatically sorted into respective departments for streamlined management.
- Staff members can review and process complaints, marking them as resolved or rejected as needed.

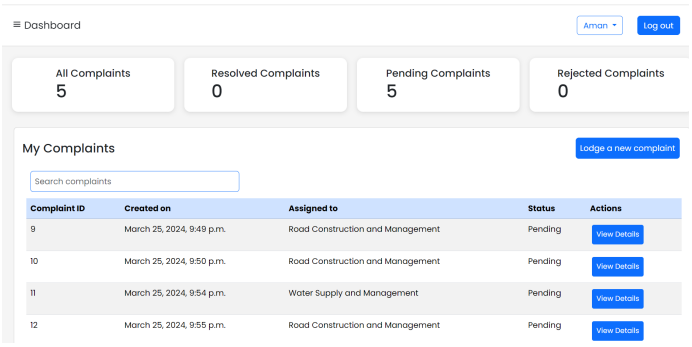

Fig. 5: User Dashboard

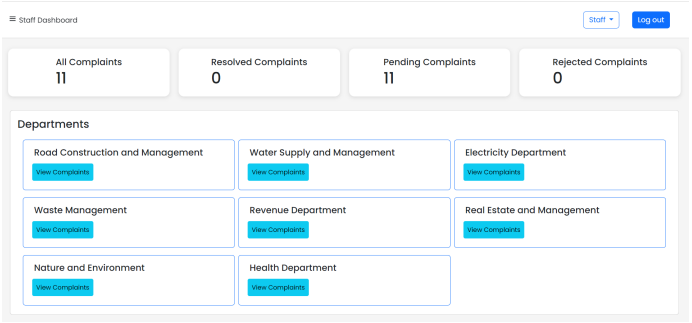

Fig. 6: Staff Dashboard

B. *User Feedback*

For user feedback, we circulated a google form along with our application. The results are as depicted in Fig. 7 and Fig. 8.

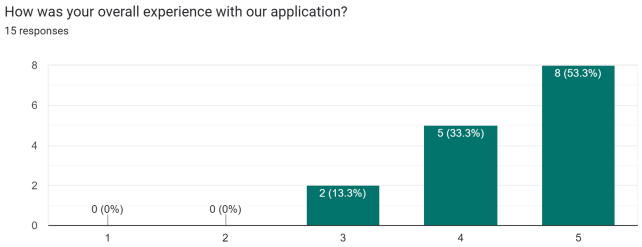

Fig. 7: Rating given by the users about our application.

IX. Future Work

In our future plans for the Multimodal complaint registration project, we’re focusing on three main areas:

1. Expanding the Dataset: To enhance the effectiveness of our complaint registration system, we’re looking to expand and diversify our dataset. This involves increasing its size by sourcing data from various platforms such as social media, news articles, and forums. Additionally, we aim to ensure diversity across demographics, cultures, and regions, with the help of expert annotators and stringent quality control measures.

2. Adding Video and Audio Classification: We’re keen on integrating multimodal capabilities into the platform. This

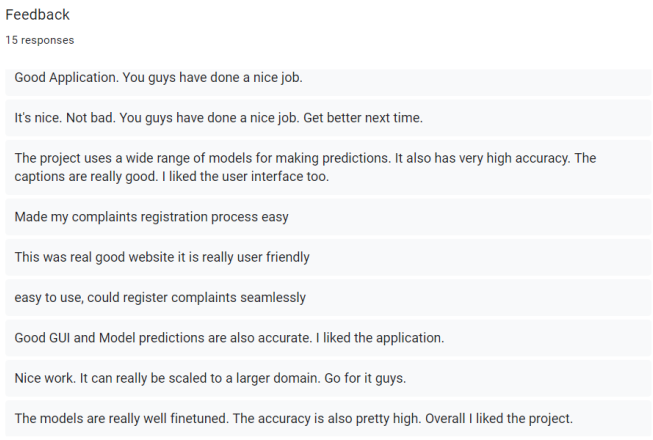

Fig. 8: User Feedback

means allowing users to submit complaints not only through text and images but also via video and audio. By implementing video and audio classification features, users can provide a richer context to their complaints, thereby enhancing the platform's overall functionality. We're also exploring techniques for cross-modal fusion to improve complaint understanding and classification accuracy.

3. Improving UI: Lastly, we're committed to enhancing the user interface (UI) to ensure a seamless and intuitive experience for our users. Through user-centric design approaches and usability testing, we aim to refine the navigation structure, improve visual elements, and optimize accessibility. This includes compliance with standards such as WCAG, catering to users with disabilities. Feedback mechanisms will be in place to gather user input for iterative UI improvements, and localization efforts will ensure inclusivity across different languages and cultural preferences. These enhancements aim to create a user-friendly and inclusive platform for complaint registration and resolution.

X. Individual Contribution

Akhilesh

1. Built an Image Classification CNN Model from Scratch.
2. Used Hugging Face Image Captioning and Text Classification Models to develop a Zero Shot Image Classification Model.
3. Collected dataset of around 800 Images grouped equally into four different classes (Road Problem, Water Problem, Electricity Problem, Waste Problem).
4. Fine Tuned a VGG16 Model for Image Classification model on the above dataset.
5. Fine tuned YOLOv8 model on the above dataset to perform Image Classification. This produced the highest Validation Accuracy (91%). Hence, it was used in the Project.

Anush

1. Developed a full-stack web application using Django, incorporating user authentication features for citizen login, and staff login, with separate citizen and staff dashboard.
2. Integrated all ML/AI models into the website for seamless complaint registration in text, image, or voice format.
3. Developed a mathematical model for implementing constraint satisfaction problem, ensuring efficient handling of complaints based on predefined criteria.
4. Developed a Naive Bayes classifier for text classification with an accuracy of 86%, aiding in the classification of text-based complaints into different categories.

Adithya

1. Built a Image Captioning model for complaints from scratch.
2. Created captions for around 800 images .
3. Fine tuned VGG-16 model for extracting image features of the images.
4. Created autoencoder and decoder model for predictiong the next word of caption.
5. Hyperparameter tuning for the LSTM model to get accuracy of 90% in generating image captions.

Abhay

1. Quality Data collection from public domains .
2. Using Various data preProcessing techniques like BOW,Word2Vec,Stemming and other NLP techniques to effectively make data ready for AI model.
3. Mapping Search problems and Developing a robust heuristics for the Model to priorities certain complaints and take into consideration emergency cases.
4. Developing a Model for voice based complaint registration which can handle more than 17 Indian local languages also gives status in the desired language.
5. Using various different machine learning approaches like Naive Bayes, SVM, and deep learning approaches like using LSTM for the multiclass classification and doing hyperparameter tuning for the same.

References

[1] L. Abisha Anto Ignatious., S. Jeevitha., M. Madhurambigai., and M. Hemalatha., "A semantic driven cnn – lstm architecture for personalised image caption generation," in *2019 11th International Conference on Advanced Computing (ICoAC)*, 2019, pp. 356–362.

[2] F. Chen and W. Qin, "Using sinaweibo microblogs to identify complaints of food customers," in *Proceedings of the 2022 3rd International Conference on Control, Robotics and Intelligent System*, ser. CCRIS '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 165–170. [Online]. Available: https://doi.org/10.1145/3562007.3562037

[3] Y. Shi, X. Li, G. Wang, and X. Jin, "Research on the recognition and classification of recyclable garbage in a complex environment based on improved yolov8s," in *2023 5th International Conference on Control and Robotics (ICCR)*, 2023, pp. 230–235.

[4] M. Gupta, A. Singh, R. Jain, A. Saxena, and S. Ahmed, "Multi-class railway complaints categorization using neural networks: Railneural," *Journal of Rail Transport Planning Management*, vol. 20, p. 100265, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210970621000317

[5] Y. Zhang, "Research on text classification method based on lstm neural network model," in *2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, 2021, pp. 1019–1022.