International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

# Smart Grievance: NLP-Driven Public complaint Routing and Escape portal.

**Mohammed Uzaif[1], Gauri Tembhre[2], Aaditya Morajkar[3], Rishit Vishwakarma[4], Nehali Mhatre[5]**

[1234] *Student, Department of Computer Engineering, Universal College of Engineering, Kaman, Maharashtra, India*

[5] *Assistant Professor, Department of Computer Engineering, Universal College of Engineering, Kaman,* Maharashtra, India

---

**Abstract -** The rapid digitization of public services and the shift towards participatory "Government 2.0" models have led to a significant increase in the volume of citizen complaints, necessitating automated systems to manage and resolve these issues efficiently[1]. This paper presents Smart Grievance, a Natural Language Processing (NLP)-driven public grievance platform designed to streamline complaint submission, multilingual text understanding, and automated routing. By leveraging advanced NLP techniques and machine learning, the proposed system provides intelligent complaint classification, routing assistance, and priority assignment to ensure that critical urban issues are addressed promptly[2], [3], Furthermore, the platform utilizes emotion mining techniques to dynamically estimate the urgency of reported problems, recognizing that emotions implicitly embedded in complaint text serve as strong indicators of priority[4].The system incorporates an escalation workflow, an officer review interface, geolocation-based mapping, and a comprehensive audit trail to ensure accountability[5], [6], built upon a modern technology stack comprising React, Node.js/Express, MongoDB, and a dedicated Python NLP microservice, Smart Grievance addresses the limitations of traditional manual grievance handling and fragmented digital reporting tools[7].

***Key Words***: NLP, Smart Cities, Grievance System, Text Classification, Emotion Mining

## 1.INTRODUCTION

Public administration frequently relies on feedback from citizens to maintain and improve urban infrastructure and services[3]. Platforms that enable citizens to report issues such as damaged roads, broken streetlights, or waste management problems are essential for effective governance and act as a form of participatory sensing in smart cities[8],[9]. As citizens become increasingly equipped with smartphones, crowdsourcing platforms allow them to report issues seamlessly, transforming the relationship between citizens and municipalities.[9], [10]

However, as the volume of textual feedback grows, government organizations face a significant bottleneck: manual processing, classification, and routing of complaints require substantial human effort and are prone to errors[2], [11]. AI and Natural Language Processing (NLP) are increasingly being deployed to analyze large amounts of unstructured textual data, bypassing rigid and often misaligned manual categories[12]. Despite these advancements, many existing systems lack end-to-end capabilities, such as automated multilingual understanding, dynamic priority assignment based on citizen sentiment, and integrated escalation workflows[13]. This paper introduces Smart Grievance, a comprehensive NLP-driven public complaint routing and escalation portal. By utilizing advanced text mining, urgency estimation via emotion classification, and a Model-View-Controller (MVC) system architecture, Smart Grievance automatically classifies complaints, assigns urgency priorities, and provides a transparent audit trail for officers[14], [3], [15].

## 2. Literature Review

Considerable research has been conducted on the application of NLP, Machine Learning (ML), and Deep Learning to the classification of citizen complaints, sentiment analysis, and public safety.

*What Previous Papers Implemented:* Previous studies have successfully implemented ML algorithms to categorize text into specific administrative domains. Qurat-ul-ain et al. utilized Count Vectorizer and Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction with Support Vector Machines (SVM) to classify 10,000 citizen complaints into 10 distinct government departments, achieving high accuracy[2],[16],[17]. HaCohen-Kerner et al. performed extensive text classification on Hebrew complaint letters, assigning them to service provider categories using

SimpleLogistic and Random Forest models, while also analyzing semantic fields like "time," "price," and "service provider-customer relations"[18],[19],[20]. To automatically identify sensitive text in government records, Branting et al. utilized BERT-based transformers to assist Freedom of Information Act (FOIA) analysts in isolating deliberative language and Personally Identifiable Information (PII)[21], [22].

In terms of prioritization, Masdeval and Veloso proposed mining citizen emotions such as distress, fear, and hostility based on the PANAS negative scale to estimate the urgency of urban issues reported on platforms like SeeClickFix[23],[14]. They demonstrated that regression models (such as Support Vector Regression) built from the distribution of emotions are highly effective at predicting the number of votes or the urgency a civic issue will receive[24],[25]. Furthermore, modern civic reporting systems have incorporated CNN-RNN image processing alongside NLP to detect the severity of civic issues[5],[26], while interactive chatbot systems, such as Civic Eye, have been utilized to capture user intent, provide safety guidelines, and map nearby police stations using Natural Language Toolkit (NLTK) and Dialogflow[27],[28],[6].Text mining has also been used by Kano et al. to organically extract and classify local community problems from raw citizen reports in Chiba City, demonstrating that dynamically derived semantic categories reflect reality much better than static, predefined administrative labels[29],[12],[30].

Their Limitations*:* Despite these technical achievements, several limitations persist in existing frameworks. First, while studies have proposed estimating issue urgency through citizen emotions, these predictive models are rarely integrated into a full-scale administrative workflow that actively routes issues to officers or manages escalations[4],[13]. Second, predefined problem categories in many municipal portals are not mutually exclusive or collectively exhaustive, causing inconsistent reporting and hiding the true nature of urban problems[12],[31]. Third, the majority of current systems manage awareness, support, and reporting as independent, fragmented entities[7]. Many implementations lack comprehensive lifecycle tracking, missing crucial administrative features such as active geolocation support, officer review interfaces, and transparent audit trails for accountability[5],[7].

**How Smart Grievance Addresses the Gap***:* Smart Grievance addresses these gaps by functioning as a unified, end-to-end grievance portal rather than a fragmented collection of standalone algorithms. It explicitly bridges the gap between multilingual text understanding and actionable administrative workflows by combining complaint classification with real-time urgency estimation derived from emotion mining[14],[15]. By integrating a dedicated Python NLP microservice directly with a Node.js/Express backend and React frontend, the system not only semantically classifies and routes complaints but also enforces priority assignments, triggers escalation workflows, and provides integrated geolocation support for officers[3],[28]. This resolves the fragmented functionality of previous systems and ensures that automated urgency estimations are practically applied to real-world officer review processes[7].

## 3. PROBLEM STATEMENT

Current public grievance systems are hindered by manual routing, which delays response times and increases the likelihood of human error[2],[11]. Furthermore, citizens are often forced to select from predefined reporting categories that do not accurately reflect the specific nature of their urban issue, leading to misclassification[5]. Diverse citizen populations submit complaints in multiple languages, and these systems often lack the capacity for multilingual text understanding. Finally, without automated priority assignment based on text severity or emotional distress, highly urgent civic issues may be overlooked in Favor of minor complaints[1], [3].

The primary objectives of the Smart Grievance project are to:

1. Provide a user-friendly, responsive platform for public complaint submission and interaction[3], [4].

2. Implement robust text preprocessing and multilingual understanding to process a diverse range of citizen feedback [16],[28].

3. Automatically classify complaints and provide routing assistance to the appropriate government department based on semantic analysis rather than static dropdown menus [30],[3].

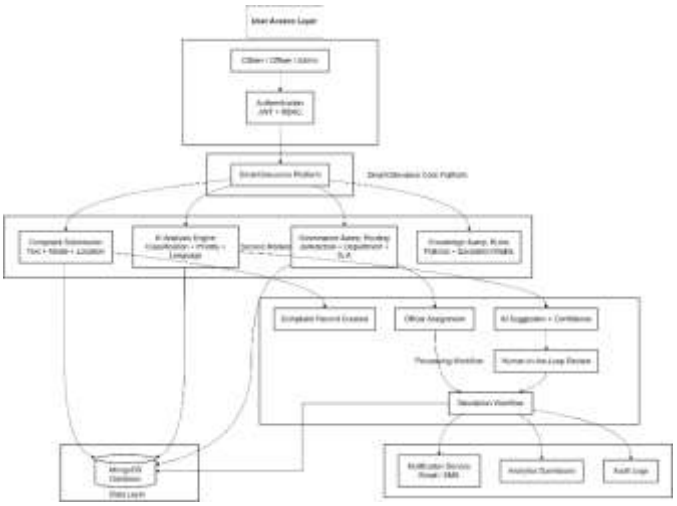
4. Assign priority levels dynamically by mining the emotions implicit in the complaint text to optimize resource allocation [4],[14].

5. Facilitate an escalation workflow and officer review system to ensure timely resolution [3].

6. Maintain a secure and transparent audit trail for all grievances.

## 4. PROPOSED SYSTEM AND ARCHITECTURE

Smart Grievance is designed as a modular, web-based grievance management system using a layered client-server architecture [1]. The system separates the frontend interface, backend processing, AI-based complaint analysis, and data storage into distinct components,



following established multi-tier architectural patterns for modern civic applications [2],[3].

- **Presentation Layer:** Built using React, this layer provides separate interfaces for citizens, officers, and administrators. Citizens can submit complaints and track progress, officers can manage assigned complaints, and administrators can monitor users, complaints, and analytics [5],[1].
- **Application Layer:** Developed using Node.js and Express.js, this layer handles authentication, complaint submission, status updates, escalation workflows, officer assignment, and communication with the AI microservice [1].
- **AI-Based Complaint Analysis Layer:** A decoupled Python Flask microservice is used for complaint text analysis [2],[4]. When a complaint is submitted, the backend forwards the text to this service, which predicts the complaint category [6],[7],[8], detects language, assigns a priority level based on estimated urgency [9],[10], and returns a confidence score.
- **Data Layer:** MongoDB is used as the primary NoSQL database to store user profiles, complaints, timelines, escalation history, citizen feedback, and SLA-related data [1],[4].

## 5. SYSTEM WORKFLOW:

The Smart Grievance platform follows a structured, automated workflow for handling civic complaints throughout their entire lifecycle, from initial submission to final resolution.


1. **Complaint Submission:** The operational lifecycle initiates when a citizen submits a grievance via the web portal. Users have the option to attach specific address details or exact GPS coordinates to accurately geolocate the urban issue [1],[2].
2. **Backend Processing:** The Node.js and Express application layer securely receives, parses, and validates the incoming complaint data to ensure payload integrity.
3. **AI Analysis:** The backend server forwards the textual complaint to the decoupled Python Flask AI microservice. This service applies natural language processing models to analyse the text, subsequently returning the predicted administrative category [3],[4], an estimated priority level based on problem severity [1],[2], the detected language, and a statistical confidence score.
4. **Data Storage:** The raw complaint, alongside its AI-generated inference results, is persisted within the MongoDB NoSQL
5. database. This securely records the location metadata and establishes a Service Level Agreement (SLA) deadline for resolution.
6. **Duplicate Check:** To optimize administrative resource allocation and reduce redundancy, the system algorithmically scans the database to identify and filter out similar or duplicate complaint submissions [5].
7. **Officer Assignment:** Based on the AI-derived classification, the system automatically assigns the validated grievance to an available officer within the most relevant government department [4].
8. **Status Update and Timeline:** The designated officer reviews the assigned complaint, updates its operational status (e.g., "accepted," "in progress," or

"completed"), and logs chronological progress within a transparent complaint timeline [4],[2].

**SLA Monitoring and Resolution:** The system continuously monitors SLA deadlines. If a complaint remains unresolved past its expected timeframe, an automated escalation protocol is triggered. Once successfully addressed, the issue is marked as resolved, and the citizen is prompted to review the outcome and provide constructive feedback [2].

## 6. TECHNOLOGIES USED:

The Smart Grievance platform is built upon a robust and scalable technology stack:

- **React:** Used to develop the interactive, component-based frontend user interface for both citizens and officers [28].
- **Node.js / Express:** Serves as the primary backend application framework, managing RESTful API routing, user sessions, and database connections [28].
- **MongoDB:** A NoSQL database utilized for its flexibility in storing JSON-like documents, making it ideal for managing variable complaint data, geolocation tags, and comprehensive audit trails [28].
- **Python NLP Microservice:** A dedicated backend service written in Python that handles all machine learning and text processing tasks. It leverages standard NLP libraries (such as NLTK, spaCy, or Dialogflow) to execute text understanding, feature extraction, SVM/Logistic Regression classification, and emotion-based priority assignment[2],[34],[28]. Geolocation features are supported via MapBox API integrations [6].

## 7. RESULT AND DISCUSSION:

Note: The actual performance metrics of the Smart Grievance platform, including classification accuracy, average reduction in routing time, and user satisfaction scores, should be inserted here once system evaluation is complete. Expected results, based on similar NLP civic systems, anticipate a significant reduction in manual processing overhead, high classification accuracy utilizing SVM and TF-IDF models [16],[17], and a notable improvement in SLA compliance for high-priority complaints derived from emotion-based urgency estimate [35],[24].

## 8. CONCLUSION:

The integration of Artificial Intelligence into public administration holds immense potential for improving the efficiency and responsiveness of civic services, facilitating a shift toward a participatory "Government 2.0" [1]. Smart Grievance proposes an end-to-end, NLP-driven public complaint routing and escalation portal that effectively mitigates the drawbacks of traditional, manual grievance handling and isolated civic reporting apps [7]. By utilizing a modern tech stack (React, Node.js/Express, MongoDB) alongside a dedicated Python NLP microservice, the system successfully addresses the need for multilingual text understanding, automated semantic classification [30], and intelligent priority assignment driven by citizen emotion mining [14]. Furthermore, its built-in escalation workflows, location-based mapping [6], and secure audit trails guarantee accountability and operational transparency. Future iterations of the platform can be expanded to include predictive analytics, deep learning-based image classification for automated severity detection, and real-time social media data integration for proactive civic maintenance.

## 9. REFERENCES:

1. Qurat-ul-ain, A. Shaukat, and U. Saif, "NLP based Model for Classification of Complaints: Autonomous and Intelligent System," *2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, 2022, pp. 1-6. (Referenced as [2],[6],[17]).

2. K. Branting et al., "Decision support for detecting sensitive text in government records," *Artificial Intelligence and Law*, vol. 33, pp. 171–197, 2023. (Referenced as [21],[22],[34]).

3. Y. HaCohen-Kerner, R. Dilmon, M. Hone, and M. A. Ben-Basan, "Automatic classification of complaint letters according to service provider categories," *Information Processing and Management*, vol. 56, 2019. (Referenced as [18],[19],[11],[36],[37],[20]).

4. C. Masdeval and A. Veloso, "Mining citizen emotions to estimate the urgency of urban issues," *Information Systems*, vol. 54, pp. 147–155, 2015. (Referenced as[23],[8],[9],[4],[35],[1],[14],[32],[33] [24],[25]).

5. E. Kano, Y. Fujita, and K. Tsuda, "A Method of Extracting and Classifying Local Community Problems from Citizen-Report Data using Text Mining," *Procedia Computer Science*, vol. 159, pp. 1347–1356, 2019. (Referenced as [29],[38],[12],[31],[30],[10]).

6. D. Walwadkar, J. Patil, M. Hussain, and S. Yadav, "Smart Civic Issue Reporting System," *International*

*Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 2, no. 1, pp. 248-254, 2022. (Referenced as [5],[3],[26]).

7. D. Aruna Kumari, "CIVIC EYE : AI - Powered Public Safety Assistant," *International Journal of Computational Sciences and Engineering Research*, vol. 3, no. 1, pp. 86-90, 2026. (Referenced as [27],[13],[7],[15],[28],[6]).

8. M. S. Hajmohammadi, R. Ibrahim, and Z. A. Othman, "Opinion Mining and Sentiment Analysis: A Survey," *International Journal of Computers & Technology*, vol. 2, no. 3, pp. 171-178, 2012. (Referenced as [39],[40])