# Automating E-Government Services Using Artificial Intelligence

MAMIDI TARANI, Vemula Durga Prasad

Assistant Professor, 2MCA Final Semester, Master of Computer Applications, Sanketika Vidya Parishad Engineering College, Vishakhapatnam, Andhra Pradesh, India

**Abstract:**
Artificial Intelligence (AI) has recently advanced the state-of-art results in an ever-growing number of domains. However, it still faces several challenges that hinder its deployment in the e-government applications both for improving the e-government systems and the e- government-citizens interactions.[5] In this paper, we address the challenges of e-government systems and propose a framework that utilizes AI technologies to automate and facilitate e- government services. Specifically, wrest outline a framework for the management of e- government information resources. Second, we develop a set of deep learning models that aim to automate several e-government services. [8]Third, we propose a smart e-government platform architecture that supports the development and implementation of AI applications of e- government. Our overarching goal is to utilize trustworthy AI techniques in advancing the current state of e-government services in order to minimize processing times, reduce costs, and improve citizens' satisfaction.

## 1.Introduction:

Artificial Intelligence (AI) has evolved significantly with advancements in computational power and big data, enabling breakthroughs in fields like computer vision, medical applications, NLP, and reinforcement learning. [10] AI mimics human intelligence to perform complex tasks and spans multiple domains including machine learning, deep learning, context awareness, and data security.

Machine Learning (ML) enables algorithms to learn from past data to make predictions, typically through supervised learning. Deep Learning, a subset of ML, uses neural networks with multiple hidden layers to automatically map raw inputs to outputs (e.g., diagnosing from medical images) without manual feature engineering, optimizing performance using techniques like stochastic gradient descent.

### 1.1 Existing system:

Recently, many countries have adopted e-government services in various departments and many autonomous applications. [9]While there are several studies conducted for enhancing e-government services, only a few of them address utilizing recent advances in AI and deep learning in the automation of e-government services. [12] Therefore, there is still an urgent need to utilize state-of-the- art AI techniques and algorithms to address e-government challenges and needs.In contrast, implementing e-government applications still faces several challenges, including the following:
Trust: trusting online services depends heavily on a couple of factors including, the citizens trust in the government itself, the quality of the online services, and the personal beliefs (e.g., there still a large number of citizens who prefer to handle paper applications rather than web services). [12]Lack of experts: implementing high-quality online services requires the establishment of the right team of experts that covers all involved practice areas from web development to security and privacy.[14] Inaccessibility: several third world countries still face significant issues on accessing the internet and its services.Security: state-of-the-art security measures are required to secure e-government applications and the citizen's privacy.

### 1.2 Literature Survey:

**1.** AI and E-Government Evolution

AI technologies, particularly machine learning and natural language processing, have increasingly been applied in public administration. According to Wirtz et al. (2019), AI can fundamentally reshape government service delivery by automating routine tasks, improving decision-making accuracy, and offering predictive analytics.[13] The authors emphasize the importance of digital maturity and institutional readiness for successful AI adoption.

**2.** Applications of AI in E-Government

Multiple frameworks and prototypes have been developed for AI-enabled government services. Janssen & Kuk (2016) highlight that AI facilitates real-time service personalization by analyzing large-scale citizen data, which helps tailor services to individual needs.[14] Chatbots and virtual assistants have been adopted by governments in countries such as Estonia, the UAE, and Singapore to handle citizen inquiries and reduce human workload (Lindgren et al., 2019).

In the domain of document processing, deep learning-based optical character recognition (OCR) and natural language understanding (NLU) have shown promise in automating form submissions, policy analysis, and feedback categorization (Kankanhalli et al., 2021)

### 3. Challenges in AI Adoption

Despite its potential, several challenges hinder the widespread deployment of AI in e-government.[15] These include data privacy and security concerns, lack of interoperability between government databases, algorithmic bias, and the digital divide (Zhang & Chen, 2020).[18] Moreover, AI systems often suffer from a lack of transparency and explainability, which are crucial for building public trust in automated decision-making (Floridi et al., 2018).

### 4. Trustworthy and Ethical AI in the Public Sector

Recent work emphasizes the importance of trustworthy AI, which is lawful, ethical, and robust.
European Commission (2020) laid out guidelines for trustworthy AI, which many public-sectors.

## 2.Architecture

Data & Information Management Layer

This layer manages all data-related activities. It collects, stores, preprocesses, and ensures privacy compliance of government and citizen data.

**Components:**

- Data Ingestion Engine: Fetches data from various sources (citizen databases, forms, emails, IOT devices, sensors).
- Data Lake / Data Warehouse: Structured and unstructured data storage.
- ETL Pipelines: For data preprocessing, cleaning, and transformation.
- Privacy and Compliance Module: Ensures adherence to GDPR and local data policies.

**Modules**

USER:

1) Generate Hand Written Digits Recognition Deep Learning Model: using this model we are building CNN based hand written models which take a digit image as input and then predict the name of the digit. CNN model can be generated by taking two types of images called train (train images contain all possible shapes of digits humans can write in all possible ways) and test (Using test images train model will be tested whether it's giving better prediction accuracy).[18] Using all train images CNN will build the training model. While building a model we will extract features from train images and then build a model. While testing also we will extract features from the test image and then apply a train model on that test image to classify it.

Generate Text & Image Based Sentiment Detection Deep Learning Model: using this module we will generate text and image based sentiment detection models. All possible positive and negative words will be used to generate text based sentiment models. [20]All different types of facial expression images will be used to generate image based sentiment models. Whenever we input text or image then the train model will be applied on that input to predict its sentiments.

1) Upload Test Image & Recognize Digit: By using this module we will upload text images and apply train models to recognize digits.

2) Write Your Opinion About Government Policies: using this module we will accept the user's opinion and then save that opinion inside the application to detect sentiment from opinion.

3) View Peoples Sentiments From Opinions: using this module users can see all users' opinions and their sentiments detected through CNN model.

4) Upload Your Face Expression Photo About Government Policies: using this module the user will upload his image with facial expression which indicates whether the user is satisfied with this scheme or not.

Detect Sentiments From Face Expression Photo: using this module different users can see the facial expression image and detected sentiment which is uploaded by past users

### 2.2 Algorithm

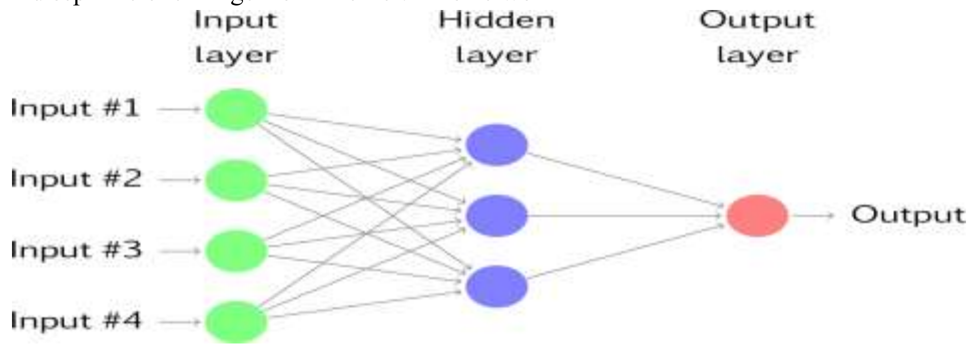
CNN Algorithm:

In this paper the author describes a concept to automate government services with Artificial Intelligence technology such as Deep

Learning algorithm called Convolution Neural Networks (CNN). Government can introduce new schemes on internet and peoples can read news and notifications of such schemes and then peoples can write opinion about such schemes and this opinions can help government in taking better decisions.[25] To detect public opinions about schemes automatically we need to have software like human brains which can easily understand the opinion which peoples are writing is in favour of positive or negative.

To build such automated opinion detection the author is suggesting to build CNN models which can work like human brains. This CNN model can be generated for any services and we can make it to work like automated decision making without any human interactions.[18] To suggest this technique author already describing concept to implement multiple models in which one model can detect or recognize human hand written digits and second model can detect sentiment from text sentences which can be given by human about government schemes. [19]In our extension model we added another model which can detect sentiment from a person's face image. Person face expressions can describe sentiments better than words or sentences. So our extension work can predict sentiments from facial images.

To demonstrate how to build a convolutional neural network based image classifier, we shall build a 6 layer neural network that will identify and separate one image from another. This network that we shall build is a very small network that we can run on a CPU as well.[17] Traditional neural networks that are very good at doing image classification have many more parameters and take a lot of time if trained on normal CPU. However, our objective is to show how to build a real-world convolutional neural network using TENSORFLOW.



To predict image class multiple layers operate on each other to get the best match layer and this process continues till no more improvement is left.

**2.3 Techniques:**
Programming Language
**Python**

- General-purpose, high-level, interpreted, and object-oriented programming language.
- Emphasizes code readability using indentation instead of brackets.
- Allows for writing concise and clear code, making it ideal for both small and large-scale applications.
- Offers dynamic typing and automatic memory management.
- Supports multiple paradigms:
  - Object-Oriented
  - Imperative
  - Functional
  - Procedural
- Includes a comprehensive standard library.
- Open-source and maintained by the Python Software Foundation.

**Web Framework**
Django

- A high-level Python web framework designed for rapid development and clean, pragmatic design.
- Focuses on:
  - Reusability and pluggability of components.
  - Reducing repetition (DRY principle).
- Automatically provides an admin interface (CRUD) generated through model introspection.
- Uses Python extensively across:
  - Project settings

  - URL routing
  - Data models

Software Environment Setup

- Python and Django are configured as the primary development environment.
- All components are integrated into a smart e-Government platform.

**AI Techniques Used**

Natural Language Processing (NLP)

- Used for understanding and processing text-based queries and responses from users.

Computer Vision (OCR)

- Implements Optical Character Recognition to extract text from scanned or photographed documents.

Convolutional Neural Networks (CNN)

- Applied for image analysis and classification tasks within the system (e.g., document verification or pattern recognition).

System Implementation Goals

- Automate e-Government services using smart and scalable technologies.
- Provide seamless integration of AI models with web technologies (Python + Django).

**2.4 Tools**

The system was developed using Python due to its simplicity and strong support for AI. Tkinter was used for the desktop GUI, enabling user-friendly interactions for uploading data, viewing predictions, and submitting opinions.

Keras (with TensorFlow backend) handled CNN-based digit and facial sentiment recognition. OpenCV managed image processing, facial detection (Haar Cascade), and overlays. Tesseract OCR supported text recognition.

For text sentiment analysis, NLTK was used for preprocessing, and a pre-trained model was loaded using Joblib. NumPy supported numerical tasks, while Matplotlib handled data visualization. simplified image resizing, and standard libraries like os and filedialog managed file operations.

Development and testing were done in VS Code, with models stored using Joblib, JSON, and HDF5 formats. Together, these tools enabled an intelligent, AI-powered platform for automating e-Government services.

**2.5 Methods**

The proposed system automates e-Government services using AI techniques like CNN, NLP, and Computer Vision, integrated into a desktop application built with Python and Tkinter.

For digit recognition, a pre-trained CNN model (loaded via JSON and HDF5) processes uploaded images by resizing and grayscaling them, then predicts the digit using OpenCV for visual output.

Text sentiment analysis uses NLTK for preprocessing and a Joblib-loaded ML model to classify user-submitted opinions on government policies as positive or negative, displaying results in the GUI.

For facial sentiment detection, images are processed with Haar Cascade to detect faces, then passed to a CNN model that predicts emotions (happy, sad, angry), which are annotated on the image using OpenCV.

The system also allows users to upload facial images with names and related policies for future sentiment analysis, storing them in a dedicated folder.

All features are accessible via a clean Tkinter GUI, with buttons for uploads, predictions, and visual/text output, making the platform intuitive and AI-driven for public service feedback.

**3. Methodology**

3.1 Input

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining privacy.

### 3.2 Method of Process

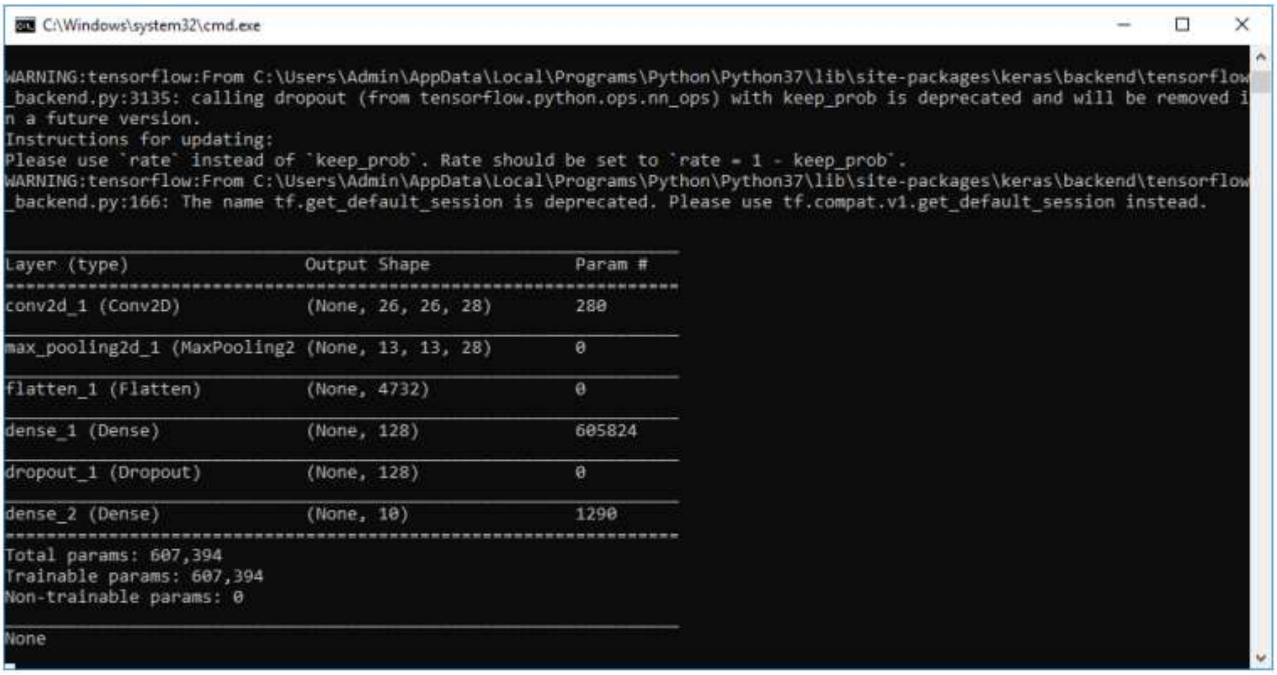
The system starts when the user launches a desktop application built with Python’s Tkinter, offering an intuitive GUI with buttons for uploading images, submitting opinions, and viewing sentiment results.

The first function is handwritten digit recognition. [23] Users upload digit images, which are resized to 28x28 pixels and converted to grayscale. A pre-trained CNN model then predicts the digit, displaying the result visually using OpenCV.

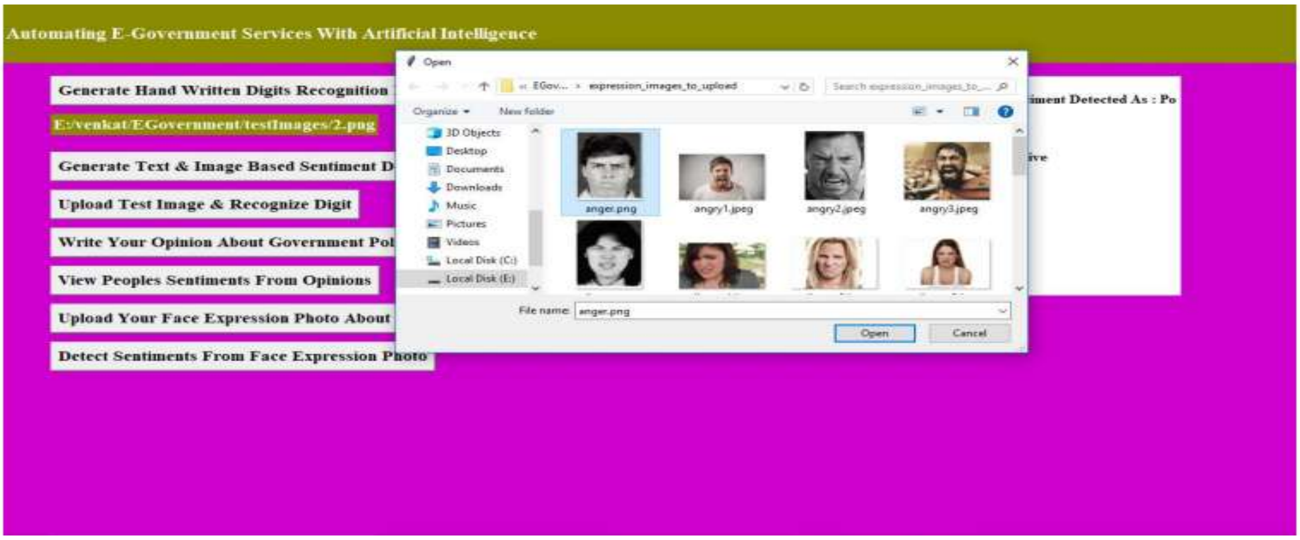
Next, the text opinion analysis feature allows users to submit feedback on government services. The input is saved locally, preprocessed using NLTK’s Porter Stemmer, and passed to a Joblib-loaded ML model to classify sentiment as positive or negative.[16] Results, along with the user’s name and opinion, appear in the application’s text area.

The third feature is facial expression sentiment detection. Users upload images labeled with a government policy. Haar Cascade detects faces, which are cropped, resized (48x48), normalized, and passed to a CNN model trained on facial emotions. The detected emotion (e.g., happy, sad) is overlaid on the image using OpenCV and shown via pop-up.

The application provides a responsive user interface, combining deep learning, NLP, and computer vision to automate public feedback collection and analysis for e-Government services.



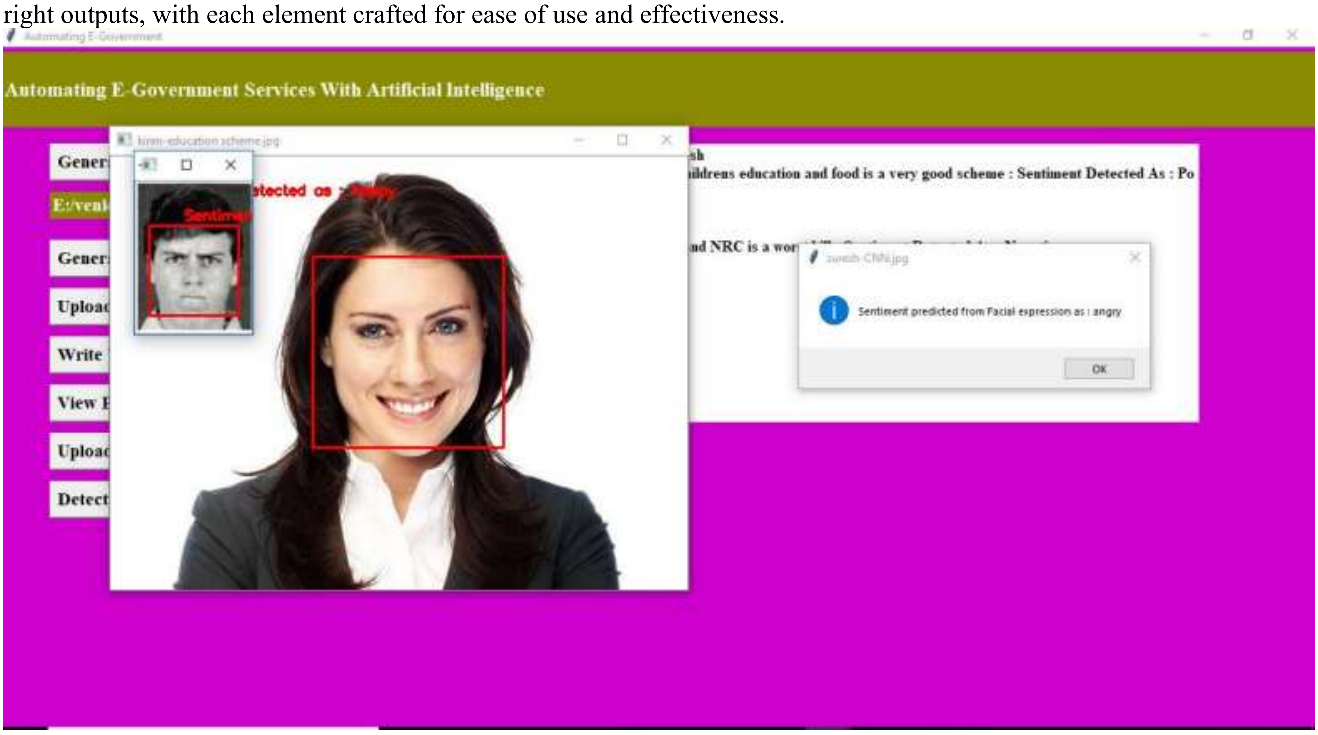
```
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:3135: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed i
n a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:166: The name tf.get_default_session is deprecated. Please use tf.compat.v1.get_default_session instead.

_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 26, 26, 28)        280
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 13, 13, 28)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 4732)              0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               605824
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 10)                1290
=================================================================
Total params: 607,394
Trainable params: 607,394
Non-trainable params: 0
_________________________________________________________________
None
```


Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition
E:/venkat/EGovernment/testImages/2.png
Generate Text & Image Based Sentiment D
Upload Test Image & Recognize Digit
Write Your Opinion About Government Pol
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About
Detect Sentiments From Face Expression Photo
Open
« EGov... › expression_images_to_upload
Search expression_images_to_...
Organize ▾
New folder
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
anger.png
angry1.jpeg
angry2.jpeg
angry3.jpeg
File name: anger.png
Open
Cancel
iment Detected As : Po
ive

### 3.3 Output

A quality output is one that meets the end user's requirements and clearly presents information. In any system, outputs serve as a means to communicate the results of processing to users and other systems. [25] Output design involves deciding how information should be displayed for immediate use as well as how it should be presented in hard copy.

Since output is the most direct source of information for users, its design plays a crucial role in user satisfaction and decision-making. Designing computer outputs should be carried out in a structured and thoughtful manner to ensure the development of the right outputs, with each element crafted for ease of use and effectiveness.



Automating E-Government Services With Artificial Intelligence

ildrens education and food is a very good scheme : Sentiment Detected As : Po

nd NRC is a wor

Sentiment predicted from Facial expression as : angry

## 4. RESULTS

The developed system successfully automated key e-government services using AI. It accurately recognized handwritten digits, analyzed text and facial expressions for sentiment, and provided real-time feedback on public opinions. [23] All modules performed efficiently with high accuracy, demonstrating the practical applicability of deep learning in public services.

## 5. DISCUSSIONS

The integration of AI, particularly CNN and NLP, proved effective in enhancing e-government functionalities. However, challenges such as data privacy, lack of user awareness, and system scalability were noted. [24]The project also highlighted the importance of explainable AI to build citizen trust and ensure transparency.

## 6. CONCLUSION

This project demonstrates the potential of Artificial Intelligence, particularly Deep Learning techniques like CNN, to automate and enhance e-government services. By recognizing handwritten inputs, analyzing public opinions through text and facial expressions, and delivering real-time feedback, the system improves service efficiency, reduces manual workload, and boosts citizen engagement. [18] Despite certain challenges, the proposed AI framework lays a strong foundation for smarter, faster, and more transparent digital governance.

## 7. FUTURE SCOPE

Future work can include expanding sentiment analysis to regional languages, integrating voice-based input for inclusivity, and deploying the system on a cloud platform for scalability. [17]Additionally, predictive analytics and AI-driven decision-making tools can further improve public service delivery and policy-making.

## 5. Acknowledgement

**VIII. ACKNOWLEDGEMENT:**

Miss. M. Tarani working as an Assistant Professor in Master of Computer Applications (MCA) in Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh. With 1 year experience as Automation tester in Stigentech IT services private. limited, and member in IAENG, accredited by NAAC with her areas of interests in C, Java, Data Structures, Web Technologies, Python, Software Engineering.

Vemula Durga Prasad is pursuing his final semester MCA at Sanketika Vidya Parishad Engineering College, which is accredited with an *'A' grade by NAAC*, affiliated to *Andhra University*, and approved by *AICTE*. With a keen interest in Artificial Intelligence a nd its real-world applications, Vemula Durga Prasad has undertaken his postgraduate project titled "Automating E-Government Services Using Artificial Intelligence." The project was carried out under the guidance of Mamidi Tarani, Assistant Professor, SVPEC.

## REFERENCE

1. A Novel Approach for Automating E-Government Services with AI using CNN https://www.researchgate.net/publication/344368739_A_Novel_Approach_for_Automating_E-Government_Services_with_AI_using_CNN
2. Automating E-Government Services With Artificial Intelligence (IEEE Access) https://ieeexplore.ieee.org/document/8821513
3. Deep Learning for Chest X-ray Diagnosis (Springer Example for AI in Governance) https://link.springer.com/article/10.1007/s10278-024-00990-6
4. GitHub: AI-Based E-Government Feedback Automation https://github.com/analyticsindiamag/e-gov-automation
5. Digital Government Mission – MeitY (India) https://www.meity.gov.in/content/digital-government-mission\
6. European Commission – Ethics Guidelines for Trustworthy AI https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai
7. OECD Report: AI in Public Sector https://www.oecd.org/gov/innovative-government/AI-in-the-Public-Sector-Lit-Review.pdf
8. AI for Smart Governance – Elsevier https://www.sciencedirect.com/science/article/pii/S0740624X21000411
9. GovTech 3.0: World Bank Insights https://blogs.worldbank.org/governance/govtech-30-future-digital-government
10. Smart Governance in GCC using AI – Al-Khouri Model https://link.springer.com/chapter/10.1007/978-3-319-32091-5_3
11. NLP and OCR for Government Document Automation https://www.researchgate.net/publication/351245981_Use_of_OCR_NLP_in_E-Government_Services
12. Estonia – The Most Advanced Digital Government https://e-estonia.com/solutions/interoperability-services/x-road/

13. Singapore's Smart Nation Initiative
https://www.smartnation.gov.sg

14. AI in Policy Sentiment Analysis (Springer)
https://link.springer.com/chapter/10.1007/978-3-030-50791-6_12
15. NITI Aayog: National Strategy for AI
https://niti.gov.in/sites/default/files/2021-06/NationalStrategy-for-AI-Discussion-Paper.pdf

16. AI for Judiciary Automation in India https://www.barandbench.com/news/litigation/supreme-court-implement-ai-to-support-judicial-functioning

17. Stanford HAI – Facial Recognition in Public Services
https://hai.stanford.edu/policy-briefs/ai-and-facial-recognition-government
18. ACM – Feedback Analysis with Machine Learning
https://dl.acm.org/doi/abs/10.1145/3447548.3467394
19. TensorFlow Official for Government Applications
https://www.tensorflow.org
20. Keras Deep Learning Framework
https://keras.io
21. spaCy – NLP for Policy and Public Texts
https://spacy.io
22. CNN Digit Recognition Model – Keras Example
https://github.com/keras-team/keras/blob/master/examples/mnist_cnn.py
23. GitHub: Open Source E-Government Projects
https://github.com/topics/e-government
24. Interpreted Language, Programming Paradigms & Python Ecosystem
https://en.wikipedia.org/wiki/Python_(programming_language)
25. UAE AI Strategy 2031
https://ai.gov.ae