iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: https://doi.org/10.22214/ijraset.2026.85541

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

AI-Driven Speech Intent Analysis for Story Generation

Dr. Kiran Kumar G H[1], Aishwarya G A[2], Akash S H[3], Akshatha G B[4], Amogh H S[5]
[1]*Associate Professor, Department of EC, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India*
[2,3,4,5]*U.G. Students, Department of EC, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India*

Abstract: ***This project explores an AI-driven system that utilizes speech intent analysis to generate personalized and context-aware stories. By leveraging natural language processing (NLP) and speech recognition technologies, the system captures spoken user input and identifies underlying intents, emotions, and keywords. These extracted features guide a generative storytelling model—powered by transformer-based architectures such as GPT or T5—to dynamically construct narratives that reflect the user's mood, preferences, and narrative direction.***

The integration of intent detection ensures stories remain relevant and coherent, transforming spontaneous speech into structured and meaningful narratives. Additionally, emotion recognition enhances story depth by adapting tone and content based on detected sentiment.

The system supports applications in education, therapy, gaming, and entertainment, enabling interactive storytelling experiences for children, learners, and users with special needs. A low-power embedded implementation using microcontrollers like the Arduino Nano 33 BLE Sense Rev2 and a digital microphone (MP34DT06JTR) demonstrates on-device processing capabilities for speech capture and intent classification. Overall, this AI-based framework bridges the gap between speech interaction and creative story generation, offering a novel and engaging human-computer interaction model.

Keywords: ***Speech intent recognition, wake-word detection, artificial intelligence, story generation, Pico-voice, Flask, Gemini, embedded system.***

I. INTRODUCTION

Traditional storytelling systems mainly depend on text inputs, predefined templates, or fixed rules. Although modern Artificial Intelligence (AI) models can generate stories from written prompts, typing every idea can sometimes make the process less convenient. Similarly, many voice-based systems are designed mainly for simple commands and have limited ability to understand the purpose of a user's request. This creates a need for a more interactive system that allows people to express their ideas naturally through speech and turn them into meaningful stories. The proposed “AI-Driven Speech Intent Analysis for Story Generation” aims to make storytelling easier, more flexible, and engaging. The system uses a digital microphone connected to the Arduino Nano 33 BLE Sense Rev2 to capture voice input. Users can activate the assistant using the wake phrase “Hey Pixel”, after which Pico-voice helps detect the wake word and recognize the intent behind the request. The recognized input is then processed through a Python Flask backend and sent to Google Gemini 2.5 Flash to generate a relevant story. Users can also enter prompts through a web-based chatbot, providing an alternative to voice interaction. The generated story is presented through the chatbot and the built-in OLED display, which also provides visual animations during system operation. This makes the interaction more interesting and allows users to view the generated content directly on the embedded device. By combining voice recognition, intent processing, AI-based story generation, and embedded technology, the project explores a convenient way to create stories based on individual ideas. It can also serve as a foundation for future applications in interactive learning, digital entertainment, and voice-assisted storytelling.

II. OBJECTIVES

The primary goals of this project are:

1) To develop a storytelling application that accepts voice and text input.
2) To activate the assistant using a predefined wake phrase.
3) To process spoken requests and identify the user’s intent.
4) To generate story content based on the user’s request using an AI model.
5) To display the generated response through the web interface and the OLED display.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

III. EXISTING SYSTEM

Most traditional storytelling platforms offer existing stories that have been written, recorded, or stored in a digital library. It allows users to choose what they want to read or listen to, but typically does not allow them to have complete control over the story's development. Those who desire a story with fewer characters, locations, or climax may need to find another story or create the changes themselves. Users of certain digital tools can construct stories by utilizing text input options. Even so, typing is still the only way to go, and it may not provide hands-free interaction like hand-to-hand contact. Basic voice-command systems are limited to a set of commands and may not fully comprehend the user's request. These limitations give way to a system that integrates voice activation, intent recognition, and AI-based story generation.

IV. PROPOSED SOLUTION

The system is designed to enable users to communicate with a storytelling assistant through speech or typed messages. If a user says “Hey Pixel,” the assistant is activated and it indicates that it’s prepared to receive... This is captured by a microphone, and the speech-processing stage helps the application hear input directly from the user (by reading data stored in the app's memory) and determine the purpose of the input. A Python Flask backend receives the request once it has been identified. It communicates with the backend of Google Gemini 2.5. Flash is a program that generates based on the user's input. Afterward, the web-based chatbot reads the generated text. A set of Arduino Nano 33 is utilized in the project. BLE Sense Rev2 and SSD1306. Displays an OLED screen that can display story text or other relevant data. Through this process, the system combines the software and embedded components of the project to provide an interactive storytelling experience.

V. METHODOLOGY

The system works through a series of connected stages as mentioned in Fig(1). It begins by receiving the user’s input, checks for the wake phrase when voice interaction is used, processes the request, generates a story, and then displays the response. The user can provide input by speaking into the microphone or typing into the chatbot interface.

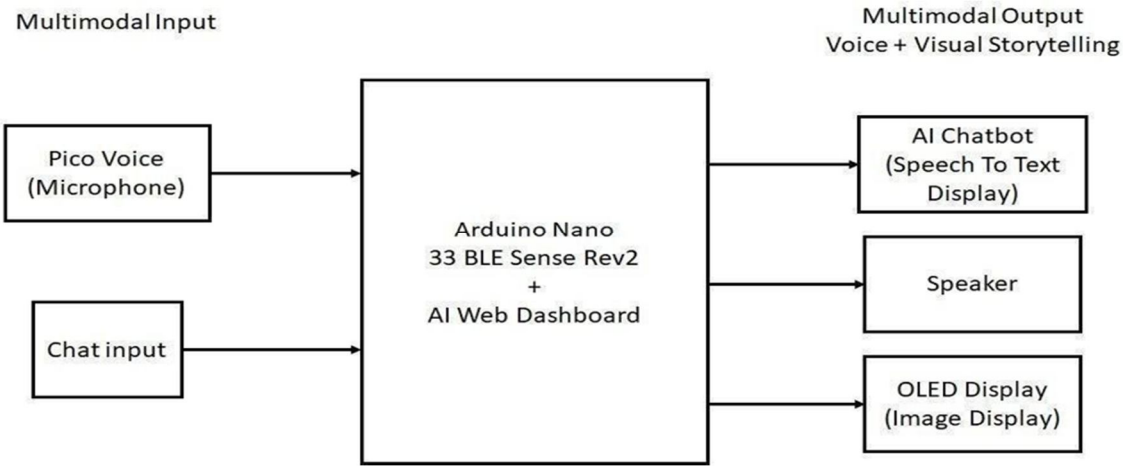


Fig.1 Block diagram.

A. Power input and Distribution stage

The three-phase power line is the first part of the system, and it is routed via a relay module for secure isolation and switching. The distribution poles, where the main voltage is accessible, get the supply from the relay. These poles serve as the focal point for electricity flow and monitoring. At this point, continuous sensing is carried out to detect any variations. This serves as the cornerstone of the entire monitoring process.

B. Voice Input and Wake-Word Detection

The microphone captures the sound when the user speaks. The assistant waits for the predefined phrase “Hey Pixel.” When this phrase is detected, the assistant becomes active and shows that it is ready to listen. The project includes an eye-opening animation and a loading indication as part of this interaction.

©IJRASET: All Rights are Reserved

C. Speech Processing and Intent Recognition

After activation, the spoken request is processed using the Pico-voice tools described in the project report. This stage helps the system interpret the user's words and identify the purpose of the request, such as asking the assistant to generate a story. The recognized request is then passed to the application for further processing.

D. Backend Processing

The Python Flask framework acts as the link between the user interface and the AI service. It receives the recognized request or the text entered by the user and manages the communication needed to generate the response.

E. AI-Based Story Generation

The user's request is sent through the backend to Google Gemini 2.5 Flash. The model uses the prompt to create story content and sends the generated text back to the application. The story is therefore based on the user's request rather than being selected from a fixed collection.

F. Output and Display

The generated story is shown in the web-based chatbot interface. The project also includes an Arduino Nano 33 BLE Sense Rev2 connected to an SSD1306 OLED display, which is used to present story text or relevant information. Push buttons are included as additional hardware input controls, as described in the project report.

VI. IMPLEMENTATION

The proposed system integrates embedded hardware and software components to perform voice-based story generation. The Arduino Nano 33 BLE Sense Rev2 manages voice input and display operations, while Pico-voice, Python Flask, and Google Gemini 2.5 Flash support speech processing and generates the story as shown in following Fig.2 and 3

A. Hardware Setup and Initialization

The Arduino Nano 33 BLE Sense Rev2 is interfaced with a digital microphone, SSD1306 OLED display, and push buttons. The microphone captures voice input, while the OLED displays animations and generated text. The hardware components are configured and programmed using the Arduino IDE.

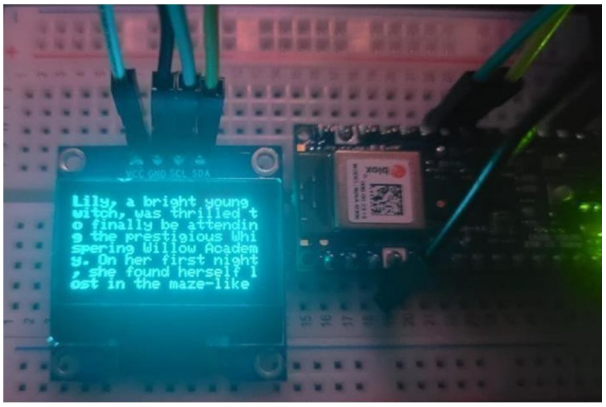


Fig.2 Story Generation

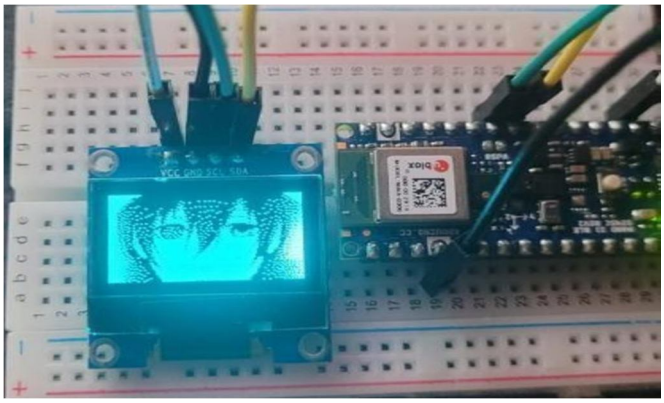


Fig.3 Eye open, Assistant is hearing

B. Wake-Word Detection and Assistant Activation

The assistant continuously monitors voice input for the wake phrase "Hey Pixel." Pico-voice Porcupine detects the phrase and activates the system. Once detected, the OLED displays an eye-opening animation to indicate successful activation.

C. Speech Intent Processing and Story Generation

After activation, the system processes the user's voice command using Pico-voice Rhino for intent recognition. When a story request is identified, the Python Flask backend manages the request and forwards the prompt to Google Gemini 2.5 Flash for story generation.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

D. Web-Based Chatbot and Embedded Display

The system also provides a web-based chatbot for entering text prompts. Python Flask processes the input and communicates with the AI model. The generated story is displayed on the chatbot, while the OLED provides embedded text and visual output.

E. Story Display and Session Completion

The generated story is presented through the available interfaces. The OLED supports scrolling to view the story in smaller sections. When the exit option is selected, the session ends and the assistant returns to its waiting state.

VII. RESULTS

The project prototype demonstrates the intended sequence of interaction. When the user says “Hey Pixel,” the assistant activates and displays an indication that it is listening. After a story request is provided, the application processes the input and obtains generated story content from the AI service. The generated response is displayed in the chatbot interface. The project report also describes the use of the OLED display to present story text or relevant information. These observations show that the main parts of the proposed system have been brought together in a working prototype. However, the project report does not include measured values for intent-recognition accuracy or response time, nor does it describe a formal user study. For this reason, this paper reports the demonstrated functions without making numerical performance claims. Further testing can be carried out to measure recognition success, response time, story relevance, and user satisfaction.

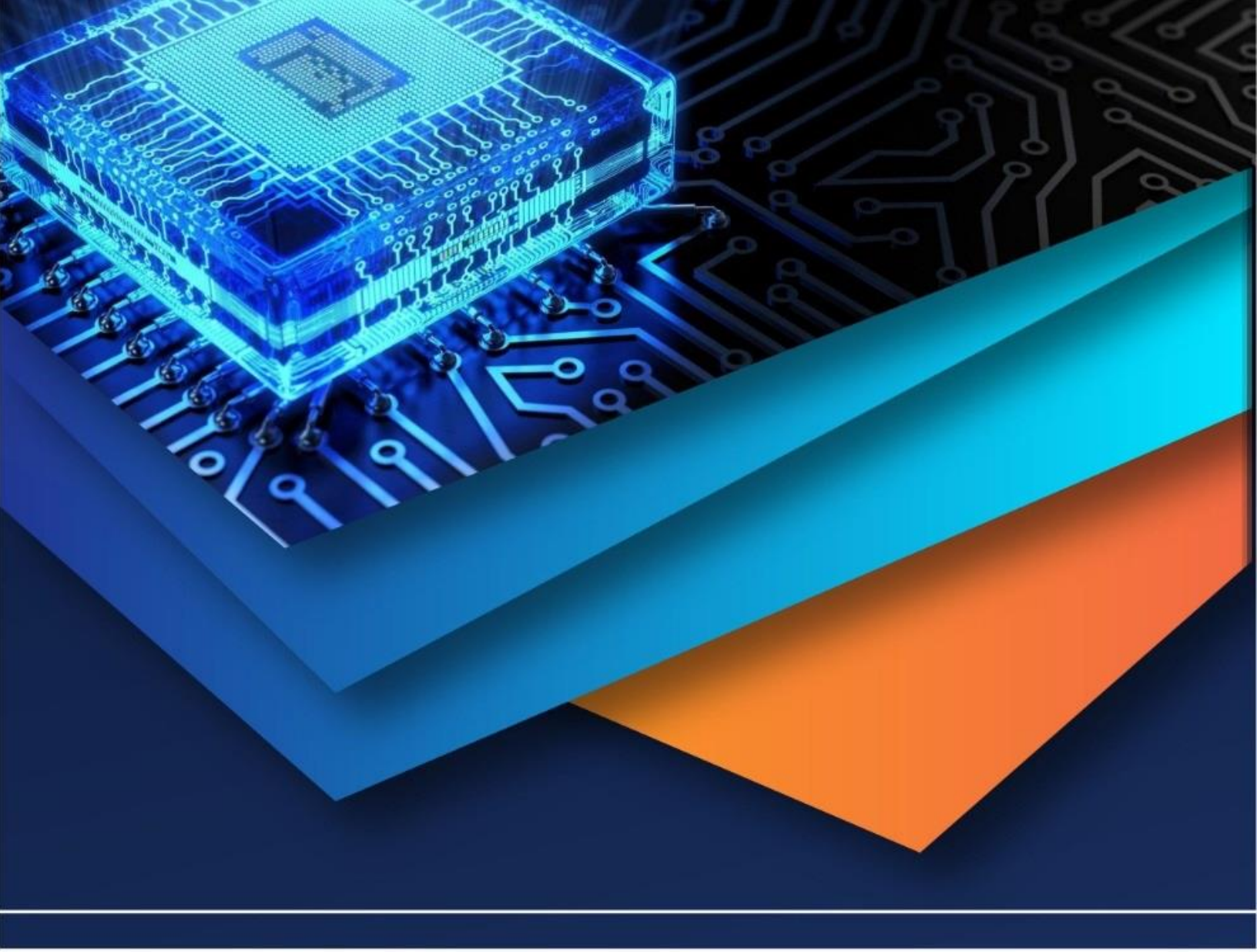
VIII. CONCLUSION

This project presents an interactive approach to storytelling by combining voice input, wake-word activation, speech intent processing, and AI-based story generation. The use of Pico-voice, Python Flask, Google Gemini 2.5 Flash, Arduino Nano 33 BLE Sense Rev2, and an OLED display brings together different software and hardware elements in one prototype. The system provides a way for users to request story content through speech or text instead of relying only on pre-written stories. In future, the project could be improved by supporting more languages and accents, retaining context across longer interactions, and evaluating its performance through systematic testing.

REFERENCES

[1] B. Liu and I. Lane, “Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling,” in Proceedings of Inter-speech 2016, pp. 685–689, 2016. doi:10.21437/Interspeech.2016-1352. 349.
[2] A. Fan, M. Lewis, and Y. Dauphin, “Hierarchical Neural Story Generation,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 889–898, 2018. doi:10.18653/v1/P18-1082.
[3] H. Rashkin, A. Choi, and J. Gao, “Plot-Machines: Outline-Conditioned Generation with Dynamic Plot State Tracking,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 4274–4295, 2020. doi:10.18653/v1/2020.emnlp-main.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)