Advanced Speech Command Recognition System

By-

Sharis Stanley Rebeiro

Shreyas Shrihari Manjula

Ankita S

Pooja Modi

***Abstract*— This paper focuses on the design and development of an advanced speech command recognition system that can process real-time voice commands and respond with the corresponding visual feedback. It allows multilingual input and has dynamic image rendering based on user commands. With a view to ensuring access and hassle-free human-computer interaction, the system has achieved a high accuracy of 98% for the recognition of commands using a Support Vector Machine model.**

Keywords—SVM Support Vector Machine

I. INTRODUCTION

Speech recognition systems form the backbone of natural language understanding and improvement of user interaction with technology. With increased dependency on voice commands to interact with devices, the need for systems that are not only efficient but also intuitive is becoming imperative. Although existing solutions have achieved certain accuracies, they usually lack dynamic feedback and do not effectively engage users. Moreover, real-time processing remains
a big challenge since the variety of input nature and immediate responses require substantial computational resources.
Natural language understanding can't be emphasized enough since it enables access to many different abilities and enhances user experience. The need for more intuitive systems, acting like a human, will drive further adoption in the consumer and enterprise spaces. The objective of the present study is to develop and implement a speech recognition system that would bridge these gaps through not only transcribing spoken commands but also linking them with actionable visual outputs. This paper proposes an advanced machine learning-based integrated system with a user-centric design that will guarantee a responsive, multilingual, and engaging interaction.

I. SYSTEM OVERVIEW

The subsequent section gives an overview of the system components and how they will interact with each other. Broadly, the system can be divided into two modules: frontend and backend. The former recognizes users' speech and renders dynamic feedback, while the latter serves to process user-issued voice commands. The Speech recognition feature makes use of JavaScript APIs in real-time. Dynamic updates have been achieved through manipulation of the DOM for an interactive experience.

At the backend, Flask API plays the role of the processing unit. It processes the speech commands as input provided by the front end, classifies these into a pre-defined set of categories, and accordingly maps the proper response. Also, the inclusion of a pre-trained SVM model improves this system for better classification accuracy of commands. The integration here points to good communication between frontend and backend and efficient carrying out of tasks.

Key Features:

1) Real-time speech-to-text, allowing for real-time feedback.
2) Multilingual support for an array of user groups. (Spanish & French)
3) It also allows dynamic rendering of images to visually confirm the commands detected.

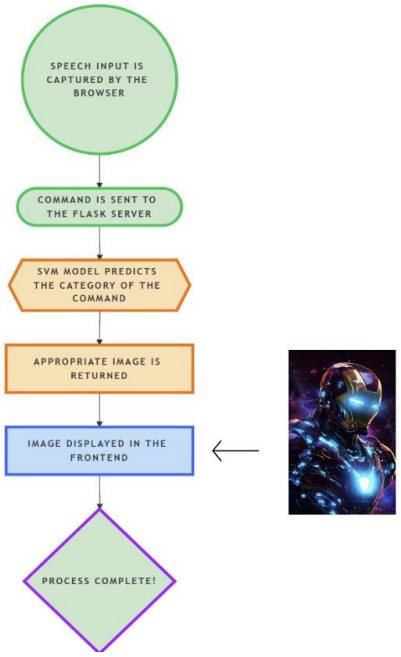
A. SYSTEM DESIGN AND WORKFLOW

The system is designed in such a way that the operation is efficient and user-friendly by having a structured workflow: data collection, preprocessing, model training, and implementation.

- Data Collection: The system starts with the collection of user commands and mapping them to predefined labels. For example, a command like "iron man" would fall under "Marvel collection." This kind of mapping lays some basis for associating spoken inputs with active responses.
- Preprocessing: Collected text data are preprocessed to turn them into a format suitable for machine learning. Using the CountVectorizer tool, textual commands are transformed into numerical vectors that allow for easy computation and pattern recognition during the training phase.
- Model Training: The system should employ a Support Vector Machine model for robust classification. Kernel trick allows the SVM to handle non-linear

relationships of high-dimensional data, making them highly effective classifiers.

- Implementation: In this case, the implementation stage should integrate all components seamlessly. The /process endpoint is the main API point used for handling the users' commands. This supports a front-end to back-end process, sending JSON based on requests and pulling results accordingly in real-time.



Key Points:

1) **Efficient Data Mapping**: Commands mapped to labels ensure structured input-output relationships.

2) **Advanced Preprocessing**: CountVectorizer transforms text into numerical vectors for machine learning.

3) **Robust Model Training**: The SVM model utilizes advanced techniques for high accuracy.

4) **API Integration**: Endpoint `/process` ensures seamless command handling and response delivery.

IV. RESULTS

The system achieved a test accuracy of 98% with the SVM model, maintaining 85% accuracy under noisy conditions. It demonstrated reliable real-time performance with minimal latency.

1) Real-Time Recognition: Speech commands are processed with a latency of ~200ms, ensuring near-instantaneous responses.

2) Multilingual Support: The system effectively switches between languages, broadening its accessibility.

3) Dynamic Image Feedback: Recognized commands trigger corresponding visual outputs, enhancing interactivity.

V. CHALLENGES AND RESOLUTIONS

The system's accuracy in noisy environments was significantly improved through a combination of advanced preprocessing techniques. These included the use of noise-reduction algorithms such as spectral subtraction and Wiener filtering, which were applied to the audio signals to minimize background noise. Additionally, feature extraction methods like Mel-Frequency Cepstral Coefficients (MFCCs) were employed to focus on the most relevant features of the speech signal, making the model more robust to ambient disturbances. Data augmentation techniques, such as adding synthetic noise to the training dataset, were also used to improve the model's ability to generalize to noisy conditions

VI. FUTURE PROSPECTS

1) Dataset Expansion:

The current system dataset consists of a small range of commands. For extending the recognition capabilities, future work will be related to the collection and integration of larger and more diverse datasets. This will include both widely used commands and domain-specific phrases to extend its application in specialized fields such as healthcare or customer service.

2) Advanced NLP Techniques:

Future models are planned to exploit state-of-the-art natural language processing methods, including transformer-based architectures, such as BERT and GPT. These advanced methods will enable the system to provide better contextual understanding of commands and handle complex queries with a higher degree of accuracy in language comprehension.

3) API Integration:

In order to make the system more dynamic and context sensitive, there is a proposal for its integration with various external APIs. For example, real-time information APIs on weather, news, or stock prices will further enable the system to answer more general queries made by users. In addition, APIs for image and video retrieval can enhance the visual feedback feature even further by making the output much more dynamic and relevant.

VII. CONCLUSION:

This advanced speech command recognition system truly represents the integration of machine learning, natural language processing, and user interface design in developing an interactive, multilingual tool. Yet, despite the challenges found, the system provides a solid framework for future improvements. Increasing the size of the datasets, using

newer NLP techniques, and also integrating third-party APIs will greatly enhance its functionality and scope of use, thus opening wider opportunities toward more intuitive human-computer interaction.

VIII. REFERNCES

[1] Cortes, C., & Vapnik, V. (1995). Support Vector Networks. Machine Learning, 20(3), 273–297.

[2] Jurafsky, D., & Martin, J. H. (2023). Speech and Language Processing. Pearson.

[3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

[4] Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach. Pearson.