EVALUATING VOICE COMMAND PIPELINES FOR DRONE CONTROL: FROM STT AND LLM TO DIRECT CLASSIFICATION AND SIAMESE NETWORKS

Lucca Emmanuel Pineli Simões
Instituto de Informática (INF)
Universidade Federal de Goiás
Goiânia, Brazil
lucca.pineli@discente.ufg.br

Lucas Brandão Rodrigues
Instituto de Informática (INF)
Universidade Federal de Goiás
Goiânia, Brazil
brandao.brandao@discente.ufg.br

Rafaela Mota Silva
Instituto de Informática (INF)
Universidade Federal de Goiás
Goiânia, Brazil
rafaelamota@discente.ufg.br

Gustavo Rodrigues da Silva
Instituto de Informática (INF)
Universidade Federal de Goiás
Goiânia, Brazil
rodrigues_da@discente.ufg.br

July 10, 2024

ABSTRACT

This paper presents the development and comparative evaluation of three voice command pipelines for controlling a Tello drone, using speech recognition and deep learning techniques. The aim is to enhance human-machine interaction by enabling intuitive voice control of drone actions. The pipelines developed include: (1) a traditional Speech-to-Text (STT) followed by a Large Language Model (LLM) approach, (2) a direct voice-to-function mapping model, and (3) a Siamese neural network-based system. Each pipeline was evaluated based on inference time, accuracy, efficiency, and flexibility. Detailed methodologies, dataset preparation, and evaluation metrics are provided, offering a comprehensive analysis of each pipeline's strengths and applicability across different scenarios.

***K*eywords** Command Mapping · Drone Control · Function Calling · LLM · NLP · Siamese Networks · Speech Recognition · STT

1 Introduction

The integration of automation and voice control in drone systems has received significant attention in recent research, driven by the need for more intuitive and efficient human-machine interaction [4, 1]. This project focuses on developing a voice command system for the Tello drone, utilizing speech recognition and deep learning models to translate voice commands into precise drone actions.

The primary challenge addressed by this project is the accurate and efficient translation of voice commands into specific drone operations. This is particularly crucial in scenarios where traditional control interfaces are impractical or where operators require hands-free operation [10, 5]. To address this challenge, we developed and evaluated three distinct pipelines. The first pipeline uses a traditional Speech-to-Text (STT) model followed by a Large Language Model (LLM) for command interpretation [11]. The second pipeline involves a direct mapping model that predicts drone commands from audio inputs without intermediate text conversion. The third pipeline employs a Siamese neural network to generalize new commands by comparing audio inputs to pre-trained examples [8].

Each pipeline was designed to balance performance, flexibility, and ease of maintenance. The methodologies employed include speech recognition techniques to convert audio to text [5], natural language processing (NLP) for command analysis [2], and neural network models for direct audio-to-command mapping and similarity-based command recognition [8]. The pipelines' effectiveness was evaluated based on accuracy, inference time, and the system's ability to generalize to new commands. The dataset for this project was prepared by recording a variety of voice commands, ensuring diversity in speech patterns and environmental conditions. Data augmentation techniques were applied to enhance model robustness. Evaluation metrics included precision, recall, F1-score, and inference time, providing a comprehensive assessment of each pipeline's performance.

This paper presents a comparative analysis of three voice command pipelines for drone control, highlighting their strengths and potential applications in various operational contexts [4, 1, 10].

2 Dataset

The purpose of the dataset is to provide comprehensive samples of voice commands for controlling the drone. Each command, such as moving the drone "right," "left," "forward," "backward," "up," and "down," was recorded multiple times to capture variations in pronunciation and intonation.

2.1 Data Augmentation

The augmentation step employed various techniques to enhance model robustness and artificially expand the dataset, making it five times larger. This expansion allowed the models to learn more effectively and generalize across diverse scenarios, thereby improving the overall performance and reliability of the voice command system. The data augmentation techniques used were:

- **Noise Addition**: Simulating different environmental sounds to mimic real-world conditions.
- **Tanh Distortion**: Utilizing Tanh activation to normalize audio signals.
- **Masking**: Applying temporal and frequency masking to obscure parts of the audio, promoting generalization.
- **Pitch-Shifting**: Altering the pitch to represent various vocal tones.

These techniques ensured that the models could generalize well across different audio conditions. After applying data augmentation, the dataset sizes for each class were increased significantly. The relevant numbers of samples for each class before and after data augmentation are shown in Table 1.

Class	Without Data Augmentation	With Data Augmentation
RIGHT	211	1055
BACKWARD	205	1025
FORWARD	202	1010
LEFT	202	1010
UP	193	965
DOWN	177	885

Table 1: Number of samples per class before and after data augmentation.

3 Methodology

The development of the voice control system for the Tello drone involved three distinct pipelines, each mapping voice commands to drone actions. All pipelines follow a general workflow: voice recording, preprocessing, model application, and function call.

Voice Recording and Processing

Audio is recorded using integrated or external microphones to ensure high-quality input. The captured audio is pre-processed to remove noise and improve signal clarity, including padding the waveform to a consistent length [3, 7].

Model Application

Each pipeline uses a different approach to interpret pre-processed audio and map it to drone commands:

- **Pipeline 1: STT and LLM** - Utilizes a Speech-to-Text (STT) model followed by a Large Language Model (LLM) for command interpretation [9].
- **Pipeline 2: Direct Model** - Employs a direct sequence classification model to predict commands from audio inputs, bypassing text conversion [5].
- **Pipeline 3: Siamese Network** - Uses a Siamese neural network to compare audio commands with pre-trained examples for command recognition [8].

Function Call

The identified or predicted command is mapped to a specific function executed by the drone [6].

3.1 Pipeline Descriptions

Pipeline 1: STT and LLM - The audio is processed and transcribed into text using the "facebook/wav2vec2-large-xlsr-53-portuguese" model. The text is then interpreted by a pretrained LLM (Llama3) to generate drone commands [5, 9]. The LLM provides responses in JSON format specifying the drone's direction, ensuring clear command interpretation.

Pipeline 2: Direct Model - Audio is directly mapped to drone commands using the same "facebook/wav2vec2-large-xlsr-53-portuguese" model, fine-tuned for this project. This pipeline eliminates the intermediate text conversion step, making it efficient but less flexible for adding new commands [5]. The model's performance is evaluated on a custom dataset of voice commands, with key metrics including classification accuracy, mean inference time, and percentage of unknown commands.

Pipeline 3: Siamese Network - A Siamese neural network is used to compare pairs of inputs to determine their similarity. The network consists of identical sub-networks that share weights and architecture, learning to project similar examples close in a latent space while dissimilar examples are projected farther apart using contrastive loss [8]. The network encodes audio files into fixed-size vectors stored in a vector database, where each vector corresponds to a command label. New voice commands are encoded into vectors and compared against stored vectors using a K-Nearest Neighbors (KNN) model for command matching [3].

3.2 Preprocessing Steps

Preprocessing involves loading audio files, standardizing waveform length, and extracting features to create input values compatible with the Wav2Vec2 model. All three pipelines use the following steps:

- **Padding**: Ensuring all waveforms have the same length by adding zeros to shorter waveforms [5].
- **Feature Extraction**: Standardizing waveform length and processing it to obtain input values for the Wav2Vec2 model [3].
- **Batch Padding**: Padding input values and labels to ensure uniformity across sequences in a batch [11].

In Pipeline 3, an additional method selects pairs of audio samples based on class similarity for comparison tasks [8].

This streamlined methodology enhances voice command recognition for the Tello drone, balancing performance, flexibility, and ease of maintenance across different approaches.

4 Results

In this section, we present the results obtained after training and evaluating the proposed pipelines. The results include accuracy, precision, recall, F1-score, and inference times for each pipeline, both before and after fine-tuning.

Table 2 provides a comprehensive summary of the performance metrics for all three pipelines. Each row represents a pipeline, while the columns contain the evaluation metrics both without and with fine-tuning.

The results indicate significant improvements with fine-tuning across all pipelines. The Direct Model pipeline achieved the highest accuracy of **0.99** after fine-tuning. In terms of inference time, the Siamese Network pipeline was the most efficient, making it highly suitable for real-time applications with an inference time of **0.006** seconds. Although it had a slightly lower accuracy of 0.74, its ability to generalize to new commands makes it a flexible option. The STT and LLM pipeline, while achieving a good balance in terms of accuracy and precision, had the longest inference time due to the sequential nature of the tasks involved.

Pipeline	Accuracy	Precision	Recall	F1-Score	Inference Time (s)
STT and LLM	0.81	0.78	0.70	0.73	1.233
Classification Model	**0.99**	**0.98**	**0.99**	**0.98**	0.021
Siamese Network	0.74	0.74	0.75	0.74	**0.006**

Table 2: Summary of Performance Metrics for All Pipelines with Fine-Tuning

These findings highlight the trade-offs between accuracy, inference time, and flexibility. The ideal choice of pipeline depends on the specific requirements of the application, whether it prioritizes precision, speed, or adaptability to new commands.

4.1 Discussion

The evaluation of the three pipelines highlights the trade-offs between accuracy, inference time, and flexibility. The STT and LLM pipeline, while highly accurate, has a longer inference time due to the sequential nature of the tasks involved. The Direct Model pipeline provides the highest accuracy and a balance between precision and efficiency, making it highly suitable for real-time applications. The Siamese Network pipeline offers the best generalization capabilities and the shortest inference time, which is advantageous in dynamic environments where new commands might be introduced.

Future work will focus on further improving the models, expanding the dataset, and exploring additional techniques to enhance the performance and reliability of the voice command system for drone control.

5 Conclusions

The development and evaluation of the three voice command pipelines for controlling the Tello drone demonstrate the effectiveness of using different approaches: STT followed by LLM, direct classification, and Siamese networks. Each pipeline has its unique strengths and potential applications, depending on the specific requirements of inference time, accuracy, efficiency, and flexibility. Through a comparative analysis, we have highlighted the trade-offs between these approaches.

The results indicate that Pipeline 1 (STT and LLM) showed high accuracy and precision, but with a longer inference time compared to other pipelines. Pipeline 2 (Direct Model) proved to have the highest accuracy and precision, with a balance between precision and efficiency. Pipeline 3 (Siamese Network) showed promise in generalizing to new commands, offering the best inference time and flexibility. These findings suggest that the ideal pipeline choice depends on the specific application context and requirements. In situations requiring high precision, Pipeline 2 is most suitable, whereas Pipeline 3 is preferable where speed and flexibility are crucial. Pipeline 1 offers a balanced solution, especially useful for applications requiring high accuracy in command interpretation.

The implementation of this voice-controlled drone system demonstrates the potential of utilizing STT, NLP, and LLM technologies to create intuitive and efficient interfaces for drones. In the future, improving models and collecting more extensive datasets can further enhance the system's performance and applicability.

References

[1] Ruben Contreras, Angel Ayala, and Francisco Cruz. Unmanned aerial vehicle control through domain-based automatic speech recognition. *Computers*, 9(3):75, 2020.

[2] J De Curtó, I De Zarza, and Carlos T Calafate. Semantic scene understanding with large language models on unmanned aerial vehicles. *Drones*, 7(2):114, 2023.

[3] Abdur Razzaq Fayjie, Amir Ramezani, Doukhi Oualid, and Deok Jin Lee. Voice enabled smart drone control. In *2017 Ninth international conference on ubiquitous and future networks (ICUFN)*, pages 119–121. IEEE, 2017.

[4] Julia Hermann, Moritz Plückthun, Aysegül Dogangün, and Marc Hesenius. User-defined gesture and voice control in human-drone interaction for police operations. In *Nordic Conference on Human-Computer Interaction*, pages 1–11, 2022.

[5] DN Krishna, Pinyi Wang, and Bruno Bozza. Using large self-supervised models for low-resource speech recognition. In *Interspeech*, pages 2436–2440, 2021.

[6] Saumya Kumaar, Toshit Bazaz, Sumeet Kour, Disha Gupta, Ravi M Vishwanath, SN Omkar, et al. A deep learning approach to speech based control of unmanned aerial vehicles (uavs). In *CS & IT Conf. Proc*, volume 8, 2018.

[7] Erica L Meszaros, Meghan Chandarana, Anna Trujillo, and B Danette Allen. Speech-based natural language interface for uav trajectory generation. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 46–55. IEEE, 2017.

[8] Yang Xie, Zhenchuan Zhang, and Yingchun Yang. Siamese network with wav2vec feature for fake speech detection. In *Interspeech*, pages 4269–4273, 2021.

[9] Shuyuan Xu, Zelong Li, Kai Mei, and Yongfeng Zhang. Core: Llm as interpreter for natural language programming, pseudo-code programming, and flow programming of ai agents. *arXiv preprint arXiv:2405.06907*, 2024.

[10] Cengizhan Yapicioğlu, Zümray Dokur, and Tamer Ölmez. Voice command recognition for drone control by deep neural networks on embedded system. In *2021 8th International Conference on Electrical and Electronics Engineering (ICEEE)*, pages 65–72. IEEE, 2021.

[11] Jing Zhao and Wei-Qiang Zhang. Improving automatic speech recognition performance for low-resource languages with self-supervised models. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1227–1241, 2022.