A Simplified Pipeline for Wakeword Creation and Deployment: Leveraging Zero-Shot Text-to-Speech and ROS2 for Robotic Systems

Alexandre Costa Ferro Filho
alexandreacff@gmail.com
Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

Daniel Ribeiro da Silva
daniel.ribeiro@discente.ufg.br
Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

José Rafael Rebêlo Teles
joserafael@discente.ufg.br
Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

Gabriel Pettro Ruotolo
gabrielruotolo@discente.ufg.br
School of Electrical, Mechanical, and Computer Engineering
Federal University of Goiás
Goiânia, Brazil

Letícia Mendes
limamendes@discente.ufg.br
Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

Marcelo Henrique Lopes Ferreira
marcelomarcelo2@discente.ufg.br
Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

Telma Woerle de Lima Soares
telma_woerle@ufg.br
AKCIT - Institute of Informatics
Federal University of Goiás
Goiânia, Brazil

***Abstract*—Wake word detection is a critical component for voice-activated robotic systems. However, the lack of diverse datasets and Text-to-Speech (TTS) models in non-English languages, such as Portuguese, poses significant challenges for developing accurate models. This paper proposes a novel and streamlined pipeline for the creation of wake word models, inspired by the OpenWakeWord framework, which leverages synthetically generated datasets to train neural models. Unlike existing approaches, our method employs a zero-shot TTS model capable of producing high-quality audio across 17 languages and a broad range of speaker styles, effectively simulating a multi-speaker environment. In contrast to OpenWakeWord, which relies on English-centric data and fixed voice banks, our approach introduces multilingual support and greater speaker variability without requiring speaker enrollment. We also detail the integration of this pipeline with the Robot Operating System 2 (ROS 2), enabling real-time robotic applications. Experimental results show that our models achieve an F1-score of up to 0.91 using purely synthetic data, demonstrating the viability and effectiveness of our method. This work highlights the potential of synthetic data generation to advance voice interaction in languages with limited resources, with a particular focus on the field of robotics in Brazil.**

***Index Terms*—Zero-Shot Text-to-Speech, ROS2, Wakeword**

I. Introduction

Human-robot interaction (HRI) is a fundamental component of most robotic applications, whether for control, communication, or automation. Specific words or phrases that activate robotic systems, known as wakewords, have become a key element of modern HRI. While many existing frameworks facilitate the training of custom wakewords, they are primarily designed for English, creating challenges for speakers of other languages, such as Portuguese. As a result, users may struggle with pronunciation, or the model's performance may degrade due to difficulties in adapting to variations in speaker accents and speech patterns. This not only limits their applicability to underrepresented languages such as Portuguese but also results in a lack of adaptability to specific linguistic needs.

To address these challenges, we introduce a flexible and efficient pipeline that combines synthetic data generation via XTTS [6] a Zero-Shot Text-to-Speech (TTS) model, training with the OpenWakeWord [3] methodology, and deployment in robotics environments using ROS 2 [11]. By eliminating reliance on pre-existing wakeword systems, our approach enables customizable solutions for underrepresented languages in HRI applications.

Our experiments show that XTTS-based models significantly outperform both the OpenWakeWord [3] baseline and the commercial Porcupine [19] system, especially when using a phoneme-based strategy for negative samples. These results validate the effectiveness of our pipeline for robust wakeword detection in low-resource linguistic settings. The source code is available at: https://github.com/Pequi-Mecanico-Home/CROS_2025.

Unlike conventional approaches, our work focuses on training a wakeword model for a less common language by leveraging synthetically generated data to compensate for the scarcity of real-world datasets. This strategy enhances adaptability to specific linguistic contexts while yielding notable performance gains. Furthermore, our approach is compatible with all 17 languages supported by the XTTS [6] zero-shot TTS model, broadening its applicability across diverse linguistic contexts.

The practical implications of this methodology are significant, particularly for promoting the inclusion of underrepresented languages in voice recognition technologies. By providing an adaptable solution, this work aims to inspire further research and applications in HRI, ultimately contributing to a more inclusive and diverse technological landscape.

979-8-3315-5288-6/25/$31.00©2025 IEEE

The methodology for training wake word detection models is detailed in Section III. The deployment process is discussed in Section IV. Experimental results, including a comparison of different approaches, are presented in Section V. Real-world considerations of our approach in the robotics field are explained in Section VI. Finally, Section VII concludes the paper and outlines future research directions.

II. Related Work

Wakeword detection is a foundational component of voice-activated systems, enabling devices to respond to specific trigger phrases. Traditional methods, such as those used in commercial platforms (e.g., Amazon Alexa, Google Assistant), rely heavily on large, curated datasets of real speech recordings [1], [2]. OpenWakeWord [3] mitigates this dependency by leveraging synthetic data generation, while also democratizing wakeword development through modular architectures and lightweight inference, but its current implementation underperforms the phonetic and prosodic nuances of Brazilian Portuguese due to limited speaker diversity and reliance on generic Text to Speech (TTS) models [7]. Recent advances in few-shot learning [5] and synthetic data generation have emerged to address data scarcity, yet their integration into end-to-end wakeword pipelines—especially for underrepresented languages—remains underexplored.

Synthetic data generation via TTS systems offers a promising solution to dataset limitations. Zero-shot TTS models, such as XTTS [6], enable high-quality speech synthesis across diverse speakers without requiring extensive training data for each voice. These models have been applied to augment automatic speech recognition (ASR) systems [8] , but their potential for wakeword-specific training—particularly in multilingual contexts—has not been fully realized. For instance, while previous work in Portuguese ASR [9] demonstrates the utility of synthetic data, it's typically oriented toward general purpose speech recognition rather than tailored wakeword detection. By leveraging zero-shot TTS [6], our approach simulates multi-speaker environments and mitigates biases inherent in small real-world datasets, a critical advancement for languages with limited resources.

In robotics, voice interaction systems often rely on modular frameworks like the Robot Operating System (ROS2) [11] for integration. Prior studies, such as [10], demonstrate ROS-based speech recognition for navigation and manipulation tasks, but these implementations typically depend on off-the-shelf wakeword detectors with limited customization. ROS2 [11], with its enhanced real-time capabilities and decentralized architecture, provides a robust foundation for deploying low-latency voice interfaces. However, existing solutions [12], [13] often separate synthetic data generation from deployment architecture, such as ROS2 [11], resulting in fragmented workflows. This gap is especially pronounced in regions like Brazil, where Portuguese-language robotic applications require tailored, resource-efficient solutions.

III. Training

A. Dataset

To evaluate the proposed approaches and validate the effectiveness of our pipeline, we selected a set of wakewords that are representative of critical commands for robotic teleoperation: ”frente” (forward), ”direita” (right), ”esquerda” (left), and ”pare” (stop). These commands were chosen due to their high relevance in navigation and control tasks, making them ideal for testing the robustness and accuracy of the wakeword detection system.

*1) **Training Data**:* Training and validation were conducted exclusively using synthetic data, generated through the XTTS [6] zero-shot Text to Speech (TTS) model, which leverages a reference speaker to produce high-quality audio with diverse speaker styles. This approach simulates a multi-speaker environment, ensuring that the model is exposed to a wide range of phonetic and prosodic variations during training. To maximize dataset diversity, we explored two synthesis techniques:

- **Single-Speaker Synthesis**: Audio samples were generated using individual reference speakers, preserving the unique characteristics of each voice.
- **Multi-Speaker Combination**: Audio samples were created by combining features from multiple reference speakers, further enhancing the dataset's diversity and robustness.

The reference speakers used to generate the XTTS [6] speech samples were derived from the CML TTS Portuguese 50 dataset [18], which provides 50 distinct speakers for training and experimentation.

For each wakeword (”frente”, ”direita”, ”esquerda”, and ”pare”), a dedicated dataset of 10,000 synthetic training samples and 1,000 validation samples was generated. Each sample includes both the wakeword (e.g., ”frente”) and its corresponding negative version (speech without the wakeword). The generation process for each dataset, considering only the synthesis phase, required approximately 1.32 hours. To generate negative samples, we employed two strategies:

- **Random Portuguese Words**: Negative samples were created using random words from the Portuguese language, ensuring variability in the dataset.
- **Phoneme-Based Synthesis**: Negative samples were generated by combining phonemes to form nonsensical words, providing a broader range of phonetic patterns without semantic meaning.

By relying solely on synthetic data for training and validation, we demonstrate the feasibility of overcoming the scarcity of annotated datasets, particularly for low-resource languages like Brazilian Portuguese. The use of XTTS [6], with its multilingual capabilities and high-quality synthesis, ensures that the dataset is not only diverse but also adaptable to other languages and applications.

*2) **Test Data**:* For the evaluation of this proposed pipeline, we created a Brazilian Portuguese wakeword dataset containing 2,329 recordings of the activation words . The samples were collected from 30 different speakers, ensuring diversity in voice characteristics and intonation. The total duration of the recordings is 40 minutes and 47 seconds, with an average length of approximately 1 second per sample. The dataset has a standard sampling rate of 16 kHz, ensuring compatibility with speech recognition models.

TABLE I
DISTRIBUTION OF RECORDINGS PER WAKEWORD

Wakeword	Number of recordings
Direita	624
Esquerda	514
Frente	597
Pare	594

This dataset was specifically developed for this work, allowing the evaluation of the wakeword recognition system in Portuguese in a realistic scenario with natural speech variations.

B. Training Process

The training process for our wakeword detection pipeline was designed to ensure robustness and generalization, particularly for the Brazilian Portuguese language. Below, we describe the key steps and methodologies employed, including the generation of synthetic data, the training of the OpenWakeWord [3] model, and the data augmentation techniques applied.

*1) **Baseline Establishment**:* To establish a baseline, we first evaluated the performance of the OpenWakeWord [3] framework using synthetic data generation pipeline. This baseline provided a reference point for comparing the effectiveness of our proposed improvements. However, the baseline results revealed significant limitations, particularly in detecting wakewords in Brazilian Portuguese, as the synthetic data generated by OpenWakeWord lacked the phonetic and prosodic diversity required for robust performance in this language.

To bridge the gap between English and Brazilian Portuguese, we initially attempted to approximate the target wakewords (”frente,” ”direita,” ”esquerda,” and ”pare”) using English phonemes. This involved generating synthetic samples with phonetically similar English words, such as ”dee-raytuh” (for ”direita”), ”iskair-duh” (for ”esquerda”), ”freynt” (for ”frente”), and ”par-ree” (for ”pare”). While this approach provided a starting point, it was insufficient for capturing the nuances of Brazilian Portuguese, highlighting the need for a more tailored solution.

*2) **General Training with Data Augmentation and OpenWakeWord-Inspired Configurations**:*

- **Synthetic Data Generation with XTTS**: We use the XTTS [6] zero-shot TTS model to generate high-quality synthetic audio samples for the target wakewords in the manner described above.
- **Data Augmentation**: To enhance the robustness of the model, we applied several data augmentation techniques, including:
  - **Room Impulse Response (RIR)**: Simulated different acoustic environments by convolving the synthetic audio with RIR recordings [14].
  - **Background Noise Mixing**: Mixed the synthetic audio with background noise from datasets such as Audioset and FMA to simulate real-world conditions [16].
  - **Pitch Shifting and Time Stretching**: Applied pitch shifting and time stretching to introduce additional variability in the training data [17].

  In addition to these, other augmentation techniques were also employed, such as equalization, distortion, and dynamic range adjustments, ensuring a diverse and robust dataset.
- **Training Parameters**: The training process was conducted using the following parameters:
  - **Batch Size**: 1024 for background noise samples, 50 for adversarial negative samples, and 50 for positive samples.
  - **Model Architecture**: A DNN-based model with a layer size of 32, chosen for its balance between performance and inference speed.
  - **Training Steps**: 50,000 steps.

The selection of these hyperparameters was also influenced by the experiments reported in OpenWakeWord, which utilise a similar amount of data to the present work, thus serving as an initial reference for effective configurations with limited datasets.

IV. DEPLOYMENT

For deployment on a real robot, we developed a system utilizing ROS2 [11] communications. Given that ROS2 is built on an abstraction layer, it facilitates a more accessible understanding, particularly for beginners, while also supporting a multi-platform environment. The system incorporates a Microphone Node for raw data extraction and preprocessing, along with a WakeWord Node that operates using the trained wakeword model. The design certifies that the Microphone Node can support multiple processes requiring microphone input without encountering channel conflicts. Consequently, the WakeWord Node exclusively accesses the audio data published by the Microphone Node via a dedicated topic.

A. *Microphone Node*

The node is designed to continuously acquire audio data from the user-specified input device. To maintain compatibility with the processing pipeline, the acquired audio is resampled from its default sampling rate to a standardized target rate matching the microphone input. The resampled data is appropriately reshaped, and published as a ROS2 [11] message within a designated ROS2 topic for further processing. This method ensures that the audio data stream can be integrated into various ROS2-based robotic applications. By standardizing the format, buffer size, and sampling rate, the node facilitates efficient data handling and interoperability

TABLE II
ACCURACY, RECALL, AND F1 RESULTS FOR DIFFERENT APPROACHES

Keyword	Approach	Positive Generation	Negative Generation	Accuracy	Recall	F1
Direita	OpenWakeWord	*	*	0.78	0.17	0.58
Esquerda	OpenWakeWord	*	*	0.89	0.52	0.81
Frente	OpenWakeWord	*	*	0.76	0.07	0.50
Pare	OpenWakeWord	*	*	0.75	0.01	0.43
Direita	XTTS	Combination	Words	0.97	0.90	0.96
Esquerda	XTTS	Combination	Words	0.92	0.65	0.87
Frente	XTTS	Combination	Words	0.93	0.73	0.90
Pare	XTTS	Combination	Words	0.88	0.65	0.83
Direita	XTTS	Combination	Phonemes	0.96	0.85	0.95
Esquerda	XTTS	Combination	Phonemes	0.94	0.74	0.90
Frente	XTTS	Combination	Phonemes	0.97	0.88	0.96
Pare	XTTS	Combination	Phonemes	0.89	0.71	0.85
Direita	XTTS	Single	Phonemes	0.95	0.83	0.94
Esquerda	XTTS	Single	Phonemes	0.96	0.84	0.95
Frente	XTTS	Single	Phonemes	0.96	0.86	0.95
Pare	XTTS	Single	Phonemes	0.88	0.69	0.84
Direita	Porcupine	*	*	0.65	0.14	0.48
Esquerda	Porcupine	*	*	0.69	0.10	0.47
Frente	Porcupine	*	*	0.65	0.14	0.48
Pare	Porcupine	*	*	*	*	*

within the processing pipeline. Additionally, the structured publication of audio data within a ROS2 [11] topic favors modularity, allowing multiple downstream components to access and utilize the information for tasks such as wakeword detection, central to our implementation, as well as speech recognition and environmental sound analysis.

B. *WakeWord Node*

Our wakeword detection system is also implemented as a ROS2 [11] node that processes incoming audio data from the previously described microphone topic. The audio signals are analyzed using our WakeWord model, which is dynamically loaded from a specified configuration path, allowing flexible model selection. The inference model continuously evaluates the incoming audio, maintaining a prediction buffer that tracks detection scores over time. When the latest score exceeds a predefined threshold the system triggers a wakeword event by publishing a Boolean message to the topic. With our approach, it enables integration with various components in any ROS2 [11] system, supporting both voice-based applications and other scenarios requiring wakeword functionality, while adhering to our deployment objectives.

V. RESULTS

The results of our experiments are presented in Table II, which compares the performance of different keyword detection approaches in terms of Accuracy, Recall, and F1-score. The approaches evaluated include OpenWakeWord [3], along with a comercial approach Porcupine [19], and several configurations of our proposed pipeline using the XTTS [6] zero-shot Text to Speech (TTS) model with different synthesis strategies for positive and negative sample generation. Below, we provide a detailed analysis of the results.

*1) **XTTS-Based Approaches**:* All XTTS-based approaches demonstrated significant improvements over the OpenWakeWord [3] baseline, showcasing the effectiveness of synthetic data generation using the XTTS [6] zero-shot TTS model. Among these, the Single + Phonemes approach, which uses single-speaker synthesis for positive samples and phoneme-based negative samples, emerged as the most robust across the tested wakewords (”frente,” ”direita,” ”esquerda,” and ”pare”). This approach achieved Accuracy ranging from 0.88 to 0.96, Recall from 0.69 to 0.86, and F1-score from 0.84 to 0.95.

The random combination of a group of speakers may not have contributed as positively as expected, indicating that one of our initial hypothesis was not exactly correct. However, the combinatorial strategy came close to the single results. There is room for further studies in order to optimize the way in which the combination is accomplished. By using phoneme-based negative samples, this approach ensures that the model is exposed to a broader range of phonetic patterns, including nonsensical combinations that do not correspond to real words. This strategy enhances the model's ability to distinguish wakewords from unrelated speech, even in challenging scenarios.

*2) **Comparison with OpenWakeWord**:* In contrast to the XTTS-based approaches, the OpenWakeWord [3] baseline achieved significantly lower performance, with Recall ranging from 0.07 to 0.52 and F1-score from 0.43 to 0.81, despite its relatively high Accuracy (0.75 to 0.89). This indicates that OpenWakeWord struggles to correctly detect wakewords in low resource languages, particularly in scenarios requiring high phonetic and prosodic diversity, such as Brazilian Portuguese.

*3) **Comparison with Porcupine**:* Unlike the XTTS-based approaches, the Porcupine [19] wake word detection system, a paid API, demonstrated lower performance across most metrics. While its accuracy ranged from 0.65 to 0.69, its recall was notably low (0.11 to 0.15), leading to F1-scores around 0.48. This suggests that Porcupine struggles with consistent wake word detection, particularly in handling the phonetic and prosodic characteristics of Brazilian Portuguese. Moreover, the API does not support short keywords, which prevented us from evaluating the keyword ”Pare”. This limitation further emphasizes Porcupine’s difficulty in handling brief wake words.

VI. Downstream applications

With the implemented architecture, we have developed a system capable of interacting in a modular fashion. As previously mentioned, the WakeWord node facilitates communication with other ROS2 [11] modules without disrupting the overall system functionality. This modularity enhances the applicability of the system across various use cases, including robot teleoperation, and even emergency scenarios.

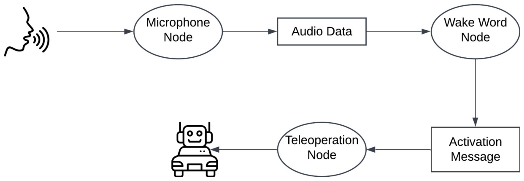

Fig. 1. Teleoperation Example

Teleoperation is a core functionality for several robotic systems, the most common approach is via Joystick or Keyboard, demanding some knowledge related to the devices.The data used in training in the previous sections address an implementation of such functionality enabled via WakeWord Detection, increasing the robot accessibility. This feature can be implemented by transmitting the output of the WakeWord Node to a Teleoperation Node that controls the mobile base.

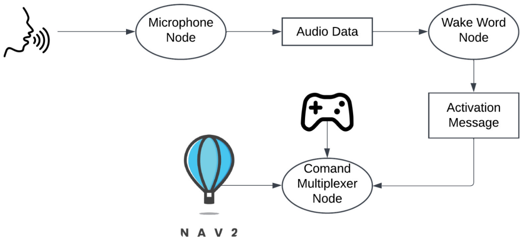

Fig. 2. Alternative Emergency Stop Example

Ensuring that robots comply with safety standards is crucial, especially in non controlled environments. In situations where an emergency stop button is not easily accessible, a wakeword can serve as an alternative mechanism to halt or redirect the robot’s movement. To achieve this, we establish communication between the WakeWord publishing node and a secondary boolean topic capable of interfacing with the robot’s mobile base. When the system detects the word ”stop” it updates the corresponding topic value, preventing the wheels from moving.

In common applications, such as domestic robotics, wakeword recognition enables service robots and voice assistants to respond naturally to human commands, thereby enhancing user interaction. In our implementation, the ASR (Automatic Speech Recognition) node is designed to communicate with the previously mentioned Microphone node. However, the ASR model only transcribes and publishes audio data after the WakeWord node confirms the detection of the target word. This procedure ensures that audio processing occurs exclusively when human interaction with the robot is explicitly detected, optimizing system efficiency and reducing unnecessary computation.

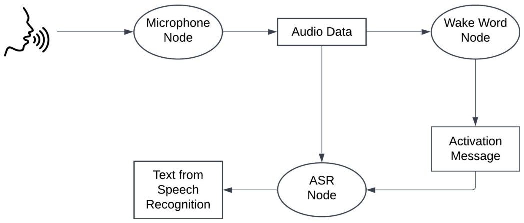

Fig. 3. ASR Triggering Example

VII. Conclusion

In this paper, we have presented a comprehensive pipeline, covering data generation, model training, and deployment, to facilitate the integration of wakeword recognition in robotics applications. Our approach provides an alternative solution for the research community, particularly for non-English-speaking regions, enabling broader accessibility and fostering further development in this field. By offering detailed instructions on data generation using Zero-Shot Text-to-Speech (TTS), training an ONNX model, and deploying it within a ROS2 [11] paradigm, we aim to expand the applicability of wakeword recognition beyond English-speaking contexts.

One of the limitations of the present approach is the linguistic coverage of XTTS [6], the state-of-the-art Zero-Shot TTS model for Portuguese. Although XTTS performs well for Portuguese, the pipeline under discussion remains constrained to the languages currently supported by XTTS. This limitation has a direct impact on the immediate applicability of the pipeline to other low-resource languages.Additionally, while the computational demands are manageable, the storage requirements for generated audio remain a key consideration.

We believe that our work contributes to making wakeword detection more accessible and adaptable, paving the way for future advancements in human-robot interaction.

Acknowledgements

We extend our gratitude to the following open-source projects and communities for their foundational contributions:
- **OpenWakeWord** for providing a flexible framework for wakeword model training and deployment.
- **Coqui TTS** for their advanced zero-shot text-to-speech tools, which enabled high-quality multilingual synthetic data generation.
- This work has been partially funded by the project Research and Development of Algorithms for Construction of Digital

Human Technological Components supported by **Advanced Knowledge Center in Immersive Technologies (AKCIT)**, with financial resources from the PPI IoT/Manufatura 4.0 / PPI HardwareBR of the MCTI grant number 057/2023, signed with EMBRAPII.

- We also thank Pequi Mecânico and our colleagues for providing the audio samples and use cases of wakewords for our research.

REFERENCES

[1] G. Chen et al., “Wake Word Detection with Streaming Transformers,” ICASSP, 2021.

[2] A. Prabhavalkar et al., “Automatic Gain Control and Multi-Style Training for Robust Small-Footprint Keyword Spotting,” Interspeech, 2015.

[3] OpenWakeWord, “An Open-Source Wakeword Detection Framework,” 2022. [Online]. Available: https://github.com/dscripka/openWakeWord

[4] K. Yao et al., “Snowboy: A Customizable Hotword Detection Engine,” IEEE MLSP, 2017.

[5] Y. Wang et al., “Few-Shot Keyword Spotting in Any Language,” Interspeech, 2022.

[6] E. Casanova et al., “XTTS: a Massively Multilingual Zero-Shot Text-to-Speech Model,” INTERSPEECH, 2024.

[7] J. Kim et al., “VITS: Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech,” ICML, 2021.

[8] R. Huang et al., “TTS-Augmented ASR for Low-Resource Languages,” IEEE SLT, 2020.

[9] L. Oliveira et al., “Synthetic Data for Portuguese ASR,” PROPOR, 2022.

[10] M. Quigley et al., “ROS-Based Voice Control for Mobile Robots,” IROS, 2016.

[11] S. Macenski et al., “ROS 2: A Flexible Architecture for Real-Time Robotics,” JOSER, 2022.

[12] J. Bohren et al., “ROS Integration for Voice-Controlled Service Robots,” ROBIO, 2018.

[13] A. Koubaa, “ROS2 for Autonomous Industrial Vehicles,” SENSORS, 2021.

[14] J. B. Allen et al., “Image Method for Efficiently Simulating Small-Room Acoustics,” JASA, 1979.

[15] J. F. Gemmeke et al., “AudioSet: An Ontology and Human-Labeled Dataset for Audio Events,” ICASSP, 2017.

[16] M. Defferrard et al., “FMA: A Dataset for Music Analysis,” ISMIR, 2017.

[17] B. McFee et al., “librosa: Audio and Music Signal Analysis in Python,” SciPy, 2015.

[18] Frederico S. Oliveira et al., “CML-TTS: A Multilingual Dataset for Speech Synthesis in Low-Resource Languages,” Springer Nature Switzerland, 2023.

[19] Picovoice, “Porcupine Wake Word Engine,” [Online]. Available: https://picovoice.ai/platform/porcupine/