

# Voice Controlled Games – The approach and challenges of implementing speech recognition and voice control in games

Dominik Strzałko  
Adam Mickiewicz University in Poznań  
Email: dominikstrzalko@gmail.com



Fundusze  
Europejskie  
Polska Cyfrowa



Rzeczpospolita  
Polska



Unia Europejska  
Europejski Fundusz  
Rozwoju Regionalnego



Udział w konferencji współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego, Program Operacyjny Wiedza Edukacja Rozwój, Priorytet III Szkolnictwo Wyższe dla gospodarki i rozwoju, Działanie 3.5 Kompleksowe programy szkół wyższych w ramach konkursu nr POWR.03.05.00-IP.08-00-PZ3/18 na Zintegrowane Programy uczelni – Ścieżka III, nr umowy POWR.03.05.00-00-Z308/18-00 Tytuł projektu: „Cyberbezpieczeństwo dla gospodarki przyszłości”

# Voice Recognition in games. Why?

Voice recognition is not a popular tool/feature among Game Developments and in Games, although We can use it for many different reasons:

- > Accessibility (Games for people with disabilities)
- > Voice Commands for units in strategy games
- > Handling the interface (for both normal and VR games)
- > Voice commands create more immersion in VR games.
- > Communication with NPC

And many more

## How it Works? Most advanced approach:

1. Recording your voice in real time
2. Recognizing words/sentences using the STT Module (eg. Azure STT)
3. Input the recognized words to Language understanding module (eg. LUIS) to detect the intents and entities
4. Transforming the output (eg. .json file) from LU Module to usable form in code (eg. Class/Object)
5. Reading the values from the final object to do certain thing in the game.
6. Action is performed in the game

This approach is going to be shown in the 3rd prototype

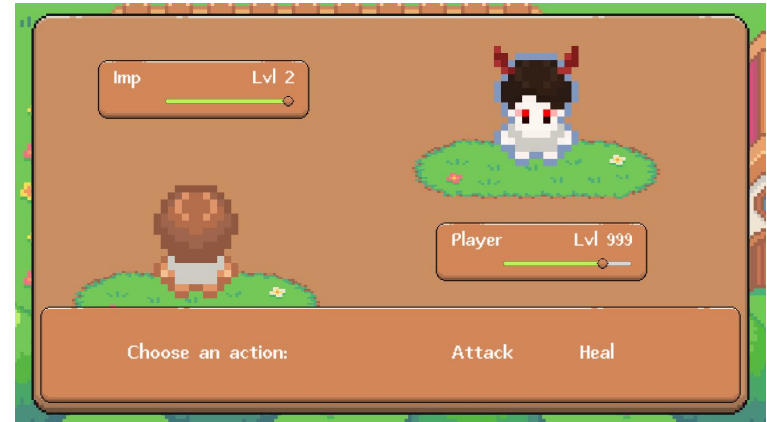
# Main challenges of my work:

- > Finding or creating modules with high accuracy in detecting voice and understanding the language
- > Implementing the modules into the Game Engines/Games
- > Creating the Plugin/Editor for game developers (For Unity/Godot)
- > Creating Prototypes that will clearly show the benefits of voice control in games (2D RPG, Quiz in Python, 3D real time Game, VR Game, Game for blind people)



# Prototypes

# 1. 2D RPG with turn based Combat [Unity 2D]



# Features

- > KeywordRecognizer supplies your app with an array of string commands to listen for

[Voice input in Unity - Mixed Reality | Microsoft Docs](#)

- > UnityEngine.Windows.Speech Namespace

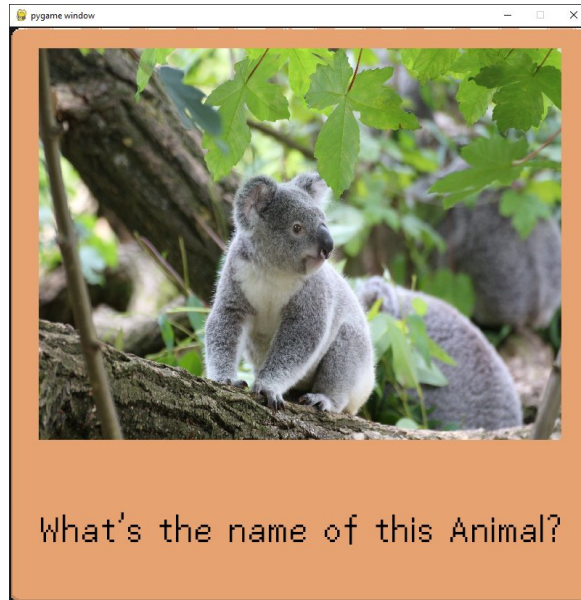
[Unity - Scripting API: KeywordRecognizer \(unity3d.com\)](#)

- > Easiest method and most popular

- > Works pretty bad

- > Its natural successor is “Cognitive Speech Services SDK” from Azure (The plugin has better Speech Accuracy results and easy access to speech-to-text decode)

## 2. SimpleAnimal Quiz with Pictures + Sounds [Python/Pygame]





# Features

- > [SpeechRecognition · PyPI](#) - Speech Recognition library
- > I'm using the Google Speech Recognition
- > Works a lot better than first prototype
- > Python Programs can be really small
- > Both Simple quiz/puzzle games and full audio games for voice assistants (like Alexa/Ciri/Amazon Echo) can be easily made using this approach
- > Pygame/Pyglet and other python based game libs are rarely used by developers, and you can't really create a more complicated games in them (or at least not that easily)

### 3. Mars landing in real time game [Unity 3D]



- App Assets
- Intents
- Entities
- Prebuilt Domains
- Improve app performance
- Review endpoint utterances
- Features
- Patterns

## Move

### Machine learning features



+ Add feature

### Examples

✓ Confirm all entities  Move to  Delete ...

Example user input



Type an example of what a user might say and hit Enter.

go 2 squares up  
 

move your ass 2 units backward  
 

go backwards 11 units  
 

move forwards 8 meters  
 

go back 1 meter  
 

## Test

Start over

Batch testing panel

Type a test utterance ...

move 4 meters up

Move (0.996)

Inspect

## Version: 0.1

Start over

Compare with published

### User input

move 4 meters up

### Top-scoring intent

Move (0.996)

[Assign to a new intent](#)

### ML entities

☐ Debug required features

MoveDistance  
4

MoveDirection  
up

### Composite entities

No predictions

### Other entities

No predictions

### Top-matched patterns

No matched patterns

### Sentiment

Enable sentiment analysis to get sentiment score

# Features

- > Most Advanced system
- > It is using the Language Understanding (LUIS) from Microsoft Azure and Microsoft speech SDK from (STT Module) Microsoft Azure

[About the Speech SDK - Speech service - Azure Cognitive Services | Microsoft Docs](#)

- > Recognizing the whole sentences pretty well and fast
- > Not that easy to implement and modify
- > Google/IBM/Amazon also have similar features (to LUIS and S-SDK)

# The End

Email: [dominikstrzalko@gmail.com](mailto:dominikstrzalko@gmail.com)

LinkedIn: <https://www.linkedin.com/in/dominik-strzałko-b48a69150/>

github: <https://github.com/SnowyCocoon>

# Free assets used in prototypes

<https://caffinate.itch.io/abaddon>

<https://ppeldo.itch.io/2d-pixel-art-game-spellmagic-fx>

<https://cypor.itch.io/12x12-rpg-tileset>

<https://retrofallgames.itch.io/pixel-ui>

<https://pythonprogramming.altervista.org/little-pygame-game-with-speech-recognition/>

“Sounds by <https://quicksounds.com>”

<https://www.pexels.com/> (Free to use (CC0) Photos of Animals)

<https://3dtextures.me/2021/05/28/rock-042/>

<https://sketchfab.com/3d-models/apollo-lunar-excursion-module-5376aae0ac484d5f8afdcae0a0d052c7>