CEDAR's PacMan: Maze Authentication Using Behavioral Biometrics
It's more than just a game...

Team Members

Colton Roach

Adeline Reichert

Jessa Gegax

Natasha Miller

Learn more about us!

Problem Statement

Objective:
- Determine the validity of a user's identity based on how they solve a maze game.

Benefits:
- More secure approach to user authentication
- Individualized to someone's cognitive patterns
- More difficult to replicate or spoof by an impostor/attacker
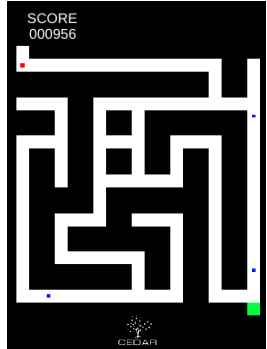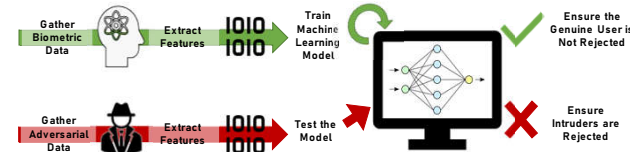

Background

Continuous authentication using behavioral biometrics is a relatively new research interest for the cybersecurity field. This is the case since the means of authentication becomes more distinct to a person's individual behavioral patterns. Other game-based authentication research has shown 80% and above accuracy using cognitive data collected after a volunteer plays a game or solves a puzzle, but higher accuracy is needed.

Methods

The maze backend collects the following data from users as they complete the challenge:
- Player's x and y position
- Number player wall collisions
- Number of player actions taken
- Player's score at each data point recorded
- Timestamp of each data point recorded
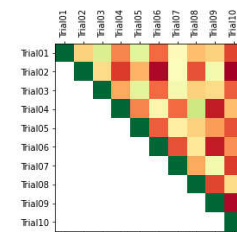
A similarity score is given through a cross-correlation approach and compared across the experiment participants.

Results

Main Goal:
- Classify maze trial runs as coming from the genuine user or as an intruder

Approach 1: MiniRocket
- Classifier designed to solve runtime concerns while producing high accuracies.
- Not extendible to one-class applications

Approach 2: Cross-Correlation:
- Able to see preliminary differences in the correlations of maze attempts by the same user vs. attempts that came from two different people.
- Requires further testing

Challenges & Future Work

The current maze is too simple to extract features that are unique enough to recognize individual users. Potential solutions include:
- Create a more difficult maze
- Introduce obstacles that require more intricate problem-solving techniques
- Consider more complex neural networks
- Reconsider MiniRocket implementation
- Reconsider feature extraction

```
import torch
import sktime
from sktime.transformations.panel.rocket._minirocket import MiniRocket as mr
```

```
mr.fit(...)          MiniRocket requires multi-class data here
mr.transform(...)
```

Advisor: Dr. Mike Borowczak (mike.borowczak@uwyo.edu)
Graduate Student Mentors:
Andey Robins (jtuttle5@uwyo.edu), Mason Johnson (mjohn107@uwyo.edu),
Shaya Wolf (swolf4@uwyo.edu)
Team Members:
Adeline Reichert (areiche1@uwyo.edu), Colton Roach (croach1@uwyo.edu),
Natasha Miller (nmille29@uwyo.edu), Jessa Gegax (jgegax@uwyo.edu)

UNIVERSITY of WYOMING

CEDAR