# Senior Project Proposal, CSEC 491

**James Zhao**

**Project Summary:** Can online learning algorithms be deployed on embedded systems at low power and cost? We investigate by creating a mind-reading algorithm inspired by Shannon's Mind Reading Paper and SEER (a sequence extractiong robot) by programming a MicroBit embedded system to be able to play the game. We will investigate this problem using both heuristic-based and reinforcement learning strategies on embedded systems, comparing performance of the two across various memory and processor constraints. We then conduct a user study to validate our approach and its performance against human opponents. Further avenues of expansion involves expanding machine learning functionality through pruning and quantization, in expanding our approach to quantify the human's level of risk-aversion, and in investigating federated learning approaches through peer-to-peer communication with other embedded systems.

**Background and Significance:** The last few years have witnessed an explosion of new developments in machine learning frameworks for embedded devices, such as TinyML and TensorFlow Lite (Warden 2020). Embedded systems are ubiquitous throughout daily life, and imbuing them with intelligence is poised to unlock advances in both portability, cost-effectiveness, and privacy when compared to cloud-based ML systems. Researchers at MIT recently demonstrated that neural networks for classification could be trained using less than 256 KB of memory (Lin 2022). This is the amount of memory available on typical off-the-shelf embedded systems such as the MicroBit (Austin 2020). We investigate whether machine learning techniques on edge devices can be deployed for a toy problem, the Matching Pennies Game, which has no pure strategy Nash equilibrium. Rather, a mixed strategy Nash equilibrium for a single trial involves both players choosing between heads and tails with equal probability. The Matching Pennies Game can be used to determine an agent's level of risk aversion through a structural econometric model (Gore 2003). We compare our machine learning approach with a baseline rule-based approach (Shannon 1954).

**Proposed Design:** At the hardware level, our approach involves porting our two algorithms (machine learning and baseline) onto a MicroBit v2 embedded microprocessor with 512 KB flash memory and 64 MHZ CPU. We develop our code and port it onto the MicroBit with ARM's Mbed operating system that can run with 4 KB RAM and 16 KB ROM, far within our system constraints (Sabri 2017). For our baseline approach, we employ a rule-based system first described by Shannon and also described by Hagelbarger with his SEER Sequence Extrapolating Robot (Hagelbarger 1956). Satat later combined both Shannon and Hagelbarger's approaches, incorporating their predictive techniques into a web app (Satat 2019). For our reinforcement learning approach, we employ the Bush Mosteller stochastic model, a reinforcement learning model designed for analyzing data with changing probabilities (Bush 1953). We follow a similar approach to Izquierdo in applying the Bush Mosteller model, formulating the model in terms of a two-player game in which each player has two options: left and right (Izquierdo 2008). We also take inspiration from Tian in leveraging cognitive hierarchy theory and Bayesian learning to beat humans in a penny-matching game (Tian 2020). In order to deploy our model on the MicroBit, we leverage the TinyML framework Tensorflow Lite for reinforcement learning (David 2021).

**Deliverables:** There are two deliverables: 1) a working embedded system that is able to proficiently play the Matching Pennies game, and 2) a write-up and supplemental code detailing our engineering approach.

**Potential Pitfalls:** One potential pitfall involves the intersecting engineering challenges of 1) training reinforcement learning algorithms to play the Matching Pennies game and 2) the difficulty of porting such complex software to run on an embedded system. To address this challenge, we plan on splitting the project into sub-components, first testing our approach on a standard personal computer, minifying it to reduce CPU and RAM usage, and then porting it onto the MicroBit.

**Schedule:**

- 1/30: Download MicroPython library and get button / display interfaces working. The library API documentation is available here: https://microbit-micropython.readthedocs.io/en/v2-docs/
- 2/6: Fully implement the rule-based algorithm following Guy Satat's write-up on Mind Reader: https://web.media.mit.edu/~guysatat/
- 2/13: Implement the reinforcement learning algorithm fully, copying the approach taken in Microbit ML: https://github.com/yumium/microbit-ML
- 2/20: Test the algorithms on the MicroBit and compare performance against the baseline.
- 2/27: Conduct a user study to validate our approach and its performance against human opponents.
- 3/6: Finalize write-up and supplemental code.

**References:**


Austin, Jonny, et al. "The BBC micro: bit: from the UK to the world." Communications of the ACM 63.3 (2020): 62-69.

Bush, Robert R., and Frederick Mosteller. "A stochastic model with applications to learning." *The Annals of Mathematical Statistics* (1953): 559-585.

David, Robert, et al. "Tensorflow lite micro: Embedded machine learning for tinyml systems." *Proceedings of Machine Learning and Systems* 3 (2021): 800-811.

Goeree, Jacob K., Charles A. Holt, and Thomas R. Palfrey. "Risk averse behavior in generalized matching pennies games." Games and Economic Behavior 45.1 (2003): 97-113.

Izquierdo, Luis R., and Segismundo S. Izquierdo. "Dynamics of the Bush-Mosteller learning algorithm in 2x2 games." *Reinforcement Learning: Theory and Applications. I-Tech Education and Publishing* (2008): 199-224.

Lin, Ji, et al. "On-device training under 256kb memory." arXiv preprint arXiv:2206.15472 (2022).

Sabri, Challouf, Lobna Kriaa, and Saidane Leila Azzouz. "Comparison of IoT constrained devices operating systems: A survey." *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2017.

Satat, Guy. “Mind Reader Final Project Report, 6.883 Online Methods in Machine Learning.” MIT Media Lab. Link: https://web.media.mit.edu/~guysatat/MindReader/GuySatat_MindReader_FinalProject.pdf. 6.883, 2019.

Shannon, Claude E. "A mind-reading machine." Bell Laboratories memorandum (1953).

Tian, Ran, et al. "Beating humans in a penny-matching game by leveraging cognitive hierarchy theory and Bayesian learning." *2020 American Control Conference (ACC)*. IEEE, 2020.

Warden, Pete, and Daniel Situnayake. *Tinyml: Machine learning with tensorflow lite on arduino and ultra-low-power microcontrollers*. O'Reilly Media, 2019.

Xiao, Perry. Designing Embedded Systems and the Internet of Things (IoT) with the ARM mbed. John Wiley & Sons, 2018.