Zach Hammitt

University of Cincinnati

Smart Hack Smasher

Smart Hack Smasher is a project that revolves around stopping malicious behavior. Specifically, this is being framed around detecting abnormalities in driverless cars. This is a fast-upcoming real world issue that is not exactly solved. By doing research and undertaking this project, it may be that a viable method for stopping dangerous behavior in driverless cars can be found. In such case, it would be great to publish a paper of this and disseminate this to the world.

Naturally being a complex issue, there is no one solution to knowing that a car is having a malicious anomaly. So in order to develop a wide ranging system that can learn over time, some form of AI system will need to be developed. Looking back there are a few courses that are rather relevant here. One thing that needs to be done is algorithm analysis. This is needed for developing, training, and optimizing the AI. Courses relevant to this are Design and Analysis of Algorithms (CS 4071) for actually designing the algorithms, AI: Principles and Applications (CS 4033) for teaching the basic concepts of how AI works, and Data Structures (CS 2028C) which did a really fine job at explaining time complexity of different operations. I will also say Data Encoding (CS 5125) really explained the root of the issue in that it showed several exploits that hackers can use to get into systems and ways you can defeat those attacks.

Classes did a pretty good job at teaching the theory side of things, but when it comes to actually doing the implementation, the work place was a very large help. What I mean by this is that the workplace taught me about a lot of the tools that exist out there that will be very useful for actually making a functional piece of software. Being a process control engineer co-op at AK Steel was largely the source of these for myself. For one I learned about using Python and within that Tensorflow. At AK, these were used to do some data analysis related to quality control issues in steel. But those same two systems are rather useful for analyzing any type of dataset and is what we will be going with for Smart Hack Smasher. Another tool that I learned about at AK that will prove to be useful is Docker. Docker was used there for modularization, the goal being keep failures as limited as possible and make them as easy as possible to recover from. This is not exactly the use case of Smart Hack Smasher, but the modularization idea is still a great one as it keeps attack vectors more limited.

As alluded to in the introduction, the primary motivation behind this project is to design something with real world applicability that can help people. Driverless cars are the future but so will be attacking them. Manipulating a driverless car could be used to commit all sorts of crimes on the passengers so doing some research into how this can be avoided could alleviate much suffering in the world. Another motivation behind the project is the desire to create an AI model. Personally, I have never made a complex AI model before. No class has had me do in depth training on one that would take months to develop. So along these lines, I am excited to learn something new and just try out something that I often hear about in the tech world.

So far Smart Hack Smasher is mostly in the preliminary phase. A problem has been identified, as has the general solution. We know that cars can be hacked and an AI model could be trained to detect and stop these attacks. The how is mostly where the problem comes in. We looked at the issue and decided a modular approach using a docker and tensorflow stack would be best after doing some research. This is mainly because how portable these systems are. Not every car manufacturer is going to use the same computer to run their systems so whatever solution cannot be bound to a single operating system. We also know that data is needed to train an AI so much work has been spent on looking at various datasets to see which one will fulfill our desire. Personally I will think that the work is done when there exists a workable simulated driving system that we can try and mess with as much as we want but the car just keeps on driving as normal. As far as if the work was quality, I suppose that comes down to if we can get our research paper to pass a peer review and get published as that would be the ultimate goal.