Three Final Pitches (only #3 changed)

August Nord

Pitch 1

Euchre AI

My plan is to create an AI to play the card game Euchre. I've played with a number of computer partners/opponents before, but have found them to be unsatisfactory. They make moves that seem illogical, and end up performing poorly. I can easily see this problem fitting into the realm of machine learning. This would likely involve experimenting with techniques like decision trees, neural networks, etc. in order to find what works well. Some research has been done in this area, so it does look like there is a dataset available to work with (https://github.com/dbravender/euchredb).

Pitch 2

Handwritten Notation Identification

Reading, writing, and editing notated music has become much easier with the development of notation softwares. This development additionally comes with the task of taking music we have in handwritten form and turning it into a digital format that is easier to read and distribute. This project would look to expand on work done in this field, trying machine learning techniques like the use of recurrent neural networks on datasets of handwritten music notation with the aim to correctly identify the various notations and symbols. (https://apacha.github.io/OMR-Datasets/)

Pitch 3

Japanese Translation

This project would look to see what can be done of translating basic Japanese sentences using a mixed method approach. Google translate is still very bad (from my personal experience) at translating Japanese to English, so I'd be curious to see what technique might lead to better translations. Machine translators are especially bad at sentences entirely in hiragana*, as word tokenization is difficult due to the lack of spaces and clear word starts**. For example, a Japanese sentence might look like わたしのともだちはいちごがすきです。Recurrent neural nets would probably be able to learn some patterns based on common words to make good guesses where the tokenization should occur. The next step would then take the words that have been translated, and attempt to organize them using search techniques. For the example sentence, after tokenization we might have got 'わたしの'、'ともだちは'、'いちごが', 'すきです'. But now, we still have a lot of options for translation. If we take them in direct sentence order and use the best word-for-word translation, we end up with "My friend strawberry like", but the preferable English sentence would be "My friend likes strawberries". Using something like beam search to keep a couple of these possible translations before picking what the best one is at the end, might help us generate a better sentence. Additionally, we probably can use

information from grammatical structures like the particles ‘の’、’は’、and ’が’ as part of the heuristic that guides us. https://kikaben.com/beam-search-for-machine-translation/.

*normally katakana and kanji are used as well, but for children and new learners who don’t know these yet, it is not unheard of to have entirely hiragana sentences
**kanji often clearly symbolizes the beginning of a word

Pitch 4

Flow Free

Flow Free (pictured below) is a phone app where you have a grid in which each pair of dots must be connected with a line that doesn’t overlap with any other lines and ideally fills the whole grid space (though I have solved some where it was possible to leave some empty space*). I could see this proposed idea either going into experimenting with some pathfinding methods to solve these puzzles, or into a way to create them with a guaranteed possible solution (*or bonus- a guaranteed single possible solution!).