PASS-CLI Improvement Plan:

This document is simply a medium to plan out improvements that are possible for the program and their feasibility.

Author: Nathan Brittin

MS2DeepScore Implementation:

* The model provided on the GitHub Repo is a Siamese neural network PyTorch model.
  + [matchms/ms2deepscore: Deep learning similarity measure for comparing MS/MS spectra with respect to their chemical similarity](https://github.com/matchms/ms2deepscore?tab=readme-ov-file)
* Since the model was built in Python it cannot be directly used by PASS-CLI as a Rust program.
* To make the model compatible with Rust programs it will have to be converted and exported from python using ONNX.
  + [ONNX with Python - ONNX 1.20.0 documentation](https://onnx.ai/onnx/intro/python.html)
  + [onnx · PyPI](https://pypi.org/project/onnx/)
* The model can then be utilized by the Rust program using the onnxruntime. This also has the benefit of possible GPU acceleration if available.
  + [onnxruntime - Rust](https://docs.rs/onnxruntime/latest/onnxruntime/)
* I know that their model can be utilized with ONNX because I see them applying it in the Galaxy distribution of their model. (Since Galaxy only uses onnx models)
  + <https://usegalaxy.eu/root?tool_id=ms2deepscore_similarity>
* I will need to do the conversion and provide it on a Zenodo repository. I am not sure whether to use their version 2 or version 3. Version 2 has a significantly smaller file size than version 3 (115MB vs 416MB).
  + I will probably go with version 3 if possible, just to stay up to date.
* Easiest way I see this being implemented is simply asking the user to download and expose the file paths to the model and settings json and then the program ingests the onnx compatible model.