AMETHYST

twosix TECHNOLOGIES

Automated Theory Substitution

Formal language generation that is explainable, explorable, and extensible by humans

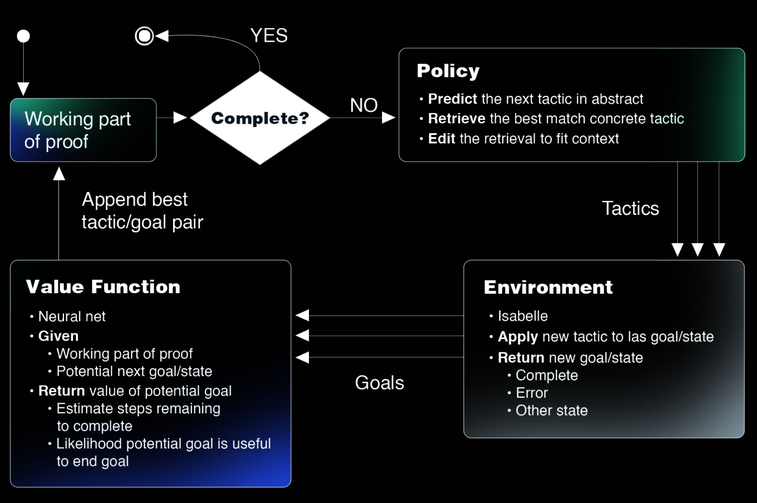
Jared Ziegler (PI), John Scebold (ML Lead)

jared.ziegler@twosixtech.com john.scebold@twosixtech.com

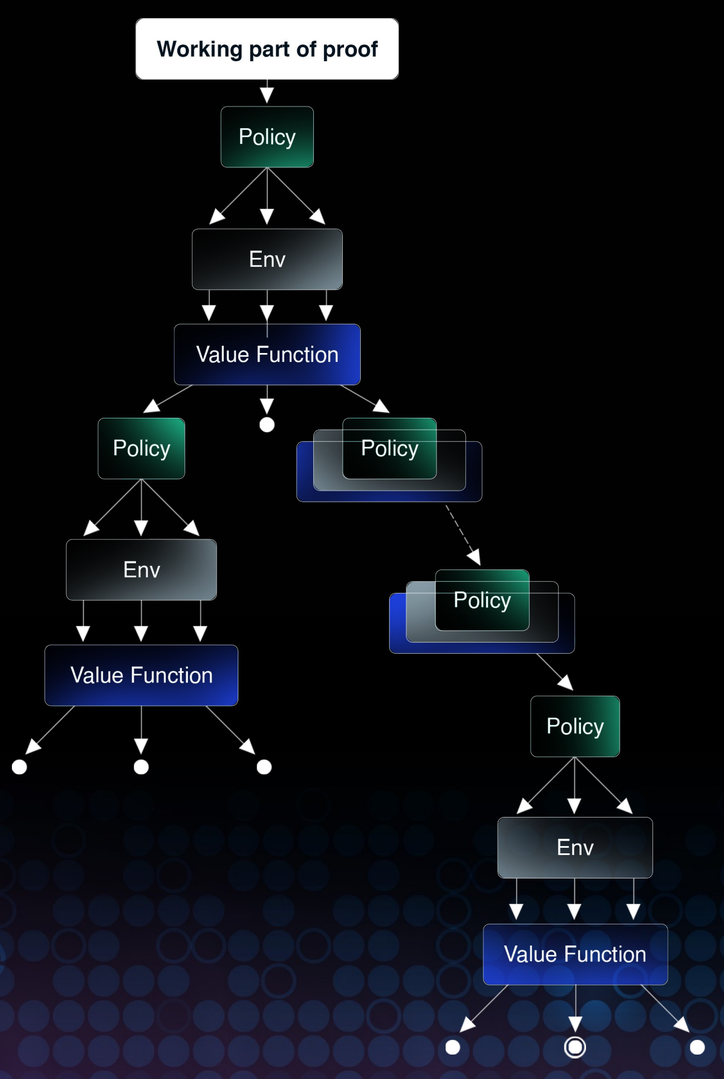
High level project description

AMETHYST is an effort to improve the state of proof repair automation using machine learning. A dataset of hand-written proofs and repairs for the Isabelle/HOL proof assistant are used to train an ensemble of machine learning models. Once trained, they can be used to generate recommended actions for repairing broken proofs at varying levels of complexity. The ensemble is orchestrated by a tree-search agent. This enables automated or human aided exploration and online learning. AMETHYST will complete proofs outright where possible or make as much progress as it can. In the second case we provide an interactive experience whereby the human receives hints to move the proof forward. One may also provide direction to get AMETHYST unstuck.

Functional system diagram of AMETHYST ensemble



The tree search agent orchestrates the modules



Coherent Generation

We were interested in LLM's capability to generate formal language.

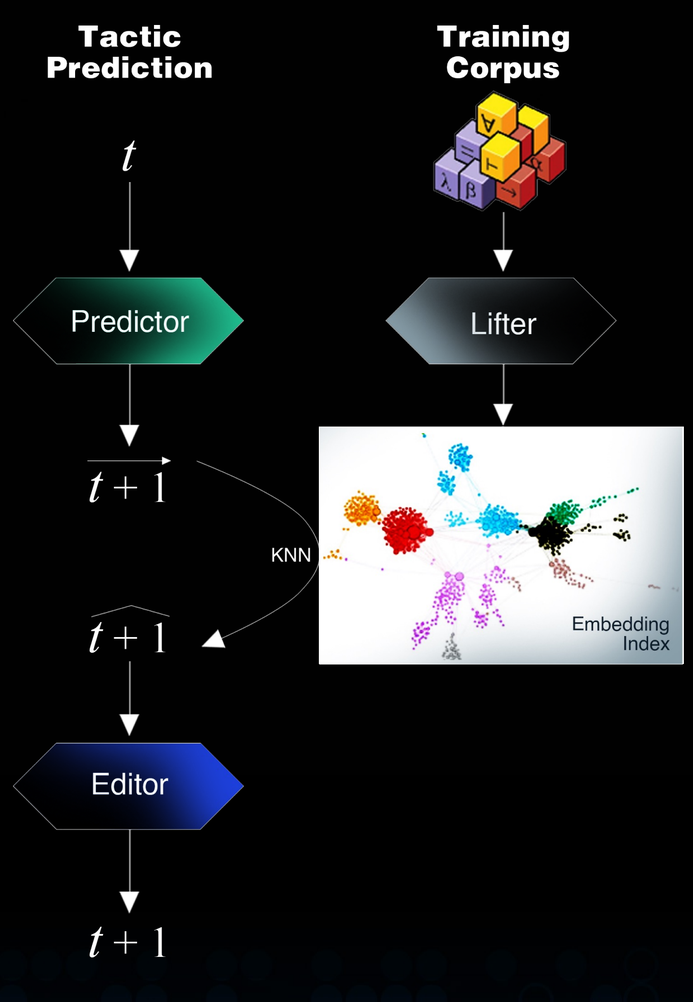
We began with HuggingFace's pretrained GPT2 and finetuned it on our corpus to outright generate tactics. It struggled to generate something Isabelle could execute.

- No tolerance for ambiguity as in natural language generation
- Relatively small training corpus
- GPT's inductive bias is too weak for the task

We employed a "retrieve and edit" strategy for tactic generation. This shifted GPT's role from generator to editor which better suits its inductive bias.

- Train a small transformer (Lifter) to embed tactics in a vector space where lexically similar texts have embeddings with high cosine similarity.
- Use the Lifter to embed the training corpus in an index.
- Train another transformer (Predictor) to take tactic text input and predict the next tactic as a vector in the embedding space. This is a KNN query to the index.
- Retrieve actual tactic text from the index.
- Use GPT2 to edit retrieved tactics to fit the context of the current proof.

Policy's retrieve-and-edit strategy comprised of three neural networks



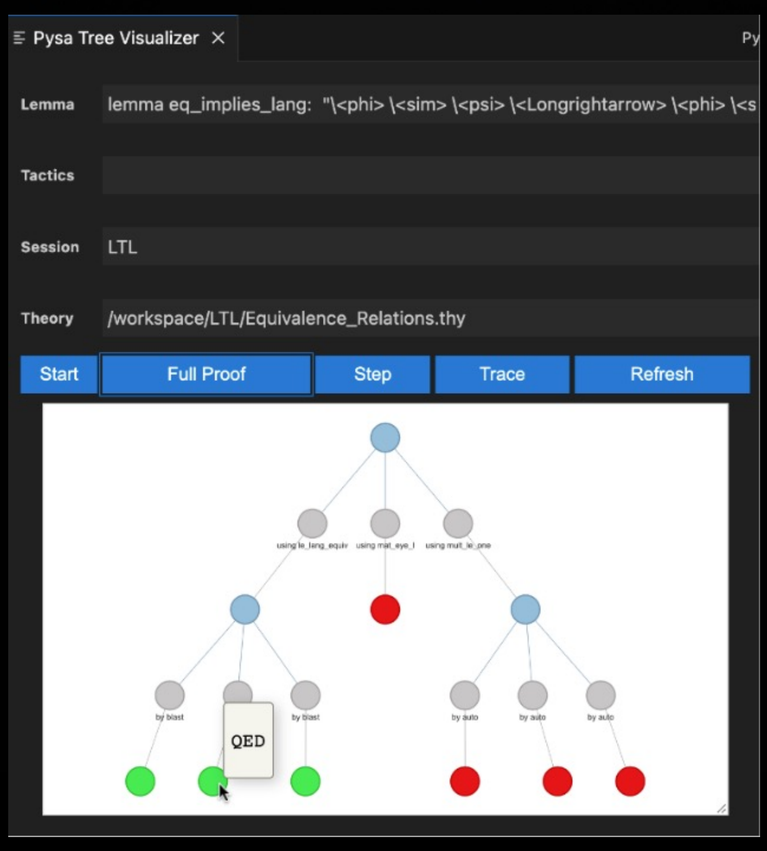
Three Xs for the Human

- Explainable – as the tree search agent automatically generates tactics and evaluates states it records everything for the human to access
  - The nearest neighbor tactics from the index along with similarity scores
  - The edits resulting from the lookups along with probability scores
  - The value function scores of the proof states resulting from tactic execution
- Explorable – the human can run the agent one step at a time or end to end in a proof search. Either way a partial or full tree is displayed graphically. The user can drag nodes, hover to view details and scores. Color coding shows tactics, proof states, and best paths.
- Extensible – we provide an interface by which the human can insert tactics anywhere in the tree to correct the agent and/or provide positive feedback on a branch of exploration. This triggers online learning.

Docker hosted web services give access to Amethyst via numerous endpoints and interaction modes.

- Python client
- Web browser
- Curl
- Visual Studio Code

User Interaction in VSCode



Computational Cybersecurity in Compromised Environments

2023 Fall Workshop | September 12-14 | Florida Institute of Technology in Melbourne, Florida