Programming Language Support for Autoassociative Memory

Abstract
This project implements a programming library for autoassociative memory that allows for easy use of structured data

Goals
- Simple, intuitive interface
- Flexible, user-defined data types to allow for a variety of use cases
- High level of parity with existing implementations

Background
- Autoassociative memories return results similar to input; useful for determining relationships between data
- Like human memory, not all inputs are stored; recall is based on learned patterns.

Motivation
- Previous models of autoassociative memories only supported simple structures, e.g. binary trees
- No common library exists for more complex recursive structures, such as parse trees

System Design

Memory Construction
- Data Type
- Training Data

Memory Usage
- Input Values
- Output Values

Post Processing

Evaluation
- Wrote recursive autoencoder for binary trees vis-à-vis Socher et. al.
- Total training error values of < 5 for data subsets using tree distance metric
- Allows for user-specified datatypes; ease of use can be aided by compiler-level introspection

Implementation
- Memory learns an encoding for input data rather than storing it
- Input structures are encoded recursively, from the bottom of the tree