Taine Zhao

twshere@outlook.com · (+86) 176-0064-7398 · thautwarm

Education

Nanjing University of Science and Technology, Jiangsu, China 09/15 – 06/19

Major: Mathematics and Applied Mathematics

Nanjing University of Science and Technology, Jiangsu, China 12/19 – Current

Major: Computer Science

Work Experience

Microsoft Research Asia, Beijing, China 11/17 – 02/19

Intern Compiler Design, Distributed System, Machine Learning Framework

- Created an implementation of dynamically loading and updating distributed data schema for Microsoft GraphEngine
- Created prototypes for MSRA internal projects about future machine learning frameworks, including gluing Python with MySQL in runtime level, and an extended SQL
- Created a LLVM IR builder framework in .NET side, which is introduced into a future/extended version of GraphEngine to achieve JIT support
- Finished a part of lowering tasks for the first/basic language in the platform of that future/extended version of GraphEngine

Personal Projects

Restrain-JIT https://github.com/thautwarm/restrain-jit

A Just-In-Time compilation for CPython

moshmosh https://github.com/thautwarm/moshmosh

A syntax extension library for CPython, bundled with the fastest implementation of pattern matching for CPython.

YAPyPy https://github.com/Xython/YAPyPy

Yet Another Python Python / Pure Python Compiler

- A pure Python compiler built on CPython providing compatibility for multiple versions of CPython 3.x
- Provided some syntax extensions like dictionary destructuring and pattern matching
- Capable of customizing parser and bytecode emitter

MLStyle.jl https://thautwarm.github.io/MLStyle.jl/latest/

A Julia package that provides ML language infrastructures like extensible pattern matching, ADTs/GADTs, etc.

- Support a large number of patterns from other languages like Haskell, Elixir, F#, OCaml, etc.
- Provided a group of concise, intuitive and convenient interfaces to customize pattern matching
- Allow users to restrict the accessing of patterns in module level
- Provided a homoiconic way to manipulation ASTs via pattern matching , which enables users to extract sub-patterns from given ASTs and rewrite them, without a prerequisite about Julia ASTs

FSTan https://github.com/thautwarm/FSTan

F# implementation of Lightweighted Higher Kined Types and type classes

- Provided a set of commonly-used type classes like Functor, Monad, MonadTrans and some instances for them
- Support ad-hoc polymorphisms(via F#'s STRT), which is an advance comparing to other implementaions

RSolve https://github.com/thautwarm/RSolve, https://github.com/thautwarm/rsolve.py

A general purposed solver for logic programming in Haskell/Python

- Established an abstraction over unification algorithms, which could be applied to many concrete instances(introduced below)
- Provided some concrete instances like HM unification and option question puzzles
- No dependency but Haskell/Python standard libraries

LanguageCollections https://github.com/thautwarm/LanguageCollections

List of languages invented by me, which has recorded my experience of learning this topic

CanonicalTraits.jl https://github.com/thautwarm/CanonicalTraits.jl

A real implementaion of Haskell's type classes in Julia

GeneralizedGenerated.jl https://github.com/thautwarm/GeneralizedGenerated.jl

A great enhancement for julia staged programming, which is widely used by the Julia community

Skills

Compiler - Front End

- experienced in creating LL(k) parsers, understand several advanced extensions and have an implementaion in Haskell: https://github.com/thautwarm/RBNF.hs
- experienced in creating lexer generators and Parser Combinators . Have several implementations in Python, F# and Haskell.
- Understand LR(1) and several of its advanced derivatives like GLR.

Compiler - Middle End

- Experienced in various kinds of (static) program analyses and transformations , e.g., implementing custom binary operators with customizable associativities and precedences, partial evaluations, forward reference resolutions, lexical/dynamic scoping analysis, syntactic macros, etc.
- Familiar with type inferences based on HM unification ,and capable of extending it with with row polymorphisms, instance resolutions, GADTs, etc.

Compiler - Back End

- Familiar with syntaxes, semantics and some intrinsics of LLVM IR
- Familiar with MIPS instructions
- Familiar with CPython bytecode instructions and code objects, etc.
- Have some experience about code generation targeting LLVM IR, CPython bytecode or MIPS ASM

Compiler - Others

- Experienced in making DSLs
- Familiar with low level data layouts
- Familiar with implementing high level language constructs (Module, Pattern Matching, Switch, Closure, etc.) for both compiled languages and interpreted languages.

Functional Programming

- Familiar with type classes and higher kinded types , and have created several implementaions
- Understand and can make good use of CPS, Y-Combinators, simple untyped/typed lambda calculus
- Understand Monad related stuffs from Monoid to MonadTrans , and have a preference of monadic coding style

Machine Learning

- Used to be familiar with commonly-used DL frameworks like PyTorch and Tensorflow, and capable of picking up again in a few minutes.
- Experienced in traditional ML toolchains like NumPy, Scikit-Learn, Pandas, Matplotlib, etc.
- Understand forward propogation and back propogation, capable of creating simple neural network frameworks
- Understand many traditional ML algorithms like KNN, K-Means, Decision Tree, Random Forest, Stacking, etc.
- 2016 CCF/DataFountain Agricultural Product Price Prediction rank 7/500+
- Have some knowledge about bioinformatics and NLP , familiar with feature extraction methods(PSSM, N-Gram, TF-IDF, etc.)
- Capable of taking advantage of ML in daily life , such as playing FGO and creating smart CLI(like lightweighted auto-jump)

Miscellaneous

- Blog: https://thautwarm.github.io/Site-32/
- PyPI: https://pypi.org/user/thautwarm/
- JuliaCN Meetup 2019: https://github.com/JuliaCN/MeetUpMaterials/tree/master/Beijing2019/thautwarm
- PyConChina 2018: http://cn.pycon.org/2018/city_beijing.html, as a lecturer(called *NightyNight*)
- PyConChina 2019: http://cn.pycon.org/2019/index.html, as a lecturer(called *thautwarm*)
- Open source contributions: contributed to some organizations such as *Microsoft, Python*
  Fetch the meaningful records from https://thautwarm.github.io/Site-32/Others/contributions.html
- Fetch the newest resume: https://raw.githubusercontent.com/thautwarm/resume/master/resume.pdf