SAND2020-7769C

MLDL

Machine Learning and Deep Learning Workshop 2020

A Whirlwind Tour of Tysons Finished and Half Finished ML/DL/RL Projects

- Tyson Bailey/05683

- Funding Sources: DOE, Professional Time

First A Joke

I like my Machine Learning like I like my data.

I ive been using machine learning for a long time.

Or Two Or Three

"Why don't eggs tell jokes?"

"I don't know. I'm not sure. I'm not sure what to say."

"Did you hear the rumor about butter?"

"No, I heard it from the other side of the room."

Abstract

When I need to take a break from my core work. I like building things. This presentation is a tour of some of those things. Some projects you might even find useful, the rest will hopefully inspire you, to share your projects, make your own, or spark an idea of your own.

Treasure Map

- Environments
  - Clue
  - Azul
  - Sharks Are Wild
  - Rubiks Cube
  - SICKE
- Tooling
  - VahVis (RL as a Service)
- Experiments
  - Command2Vec
  - Space

Environments

One of the challenges with applying reinforcement learning is simply having/building environments.

Practice building environments and understanding how to make them consistent and easy to swap out with different algorithms.

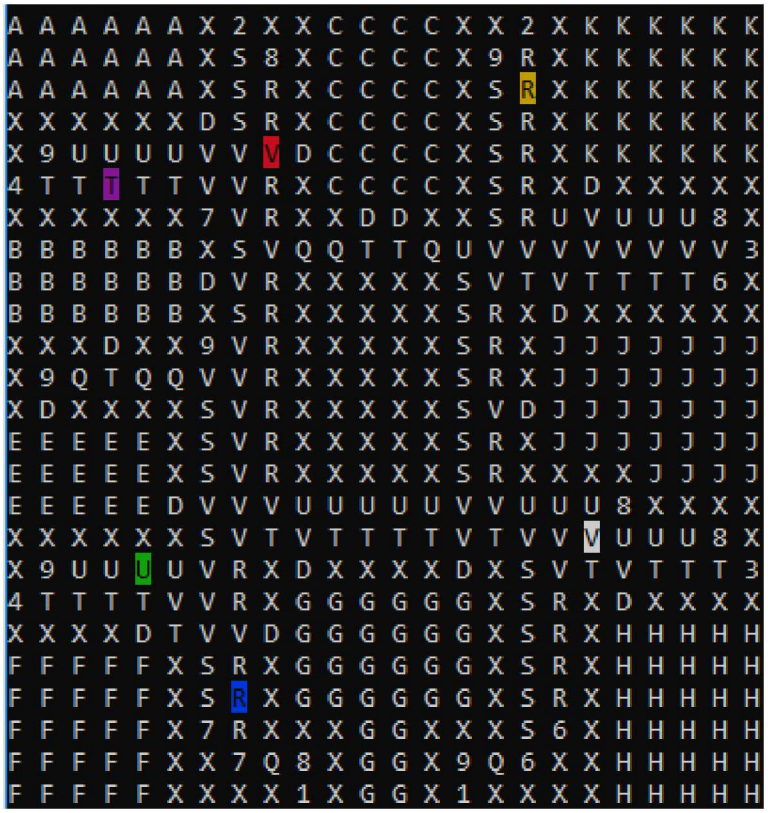
Structured so we can insert Reinforcement Learning Agents, Random Action Agents, or Human Agents allowing teamwork play.

Some of these projects are reasonably complete, some are less than complete but at least have the structure in place.

Almost all of the following links will require access, I'm happy to give access (within SNL) just shoot me an email. tbailey@sandia.gov

Clue

https://gitlab.sandia.gov/gym/clue



https://ccsearch.creativecommons.org/photos/692a324a-44a8-4282-bd2d-d63291d0eb93

Azul

https://gitlab.sandia.gov/gym/azul

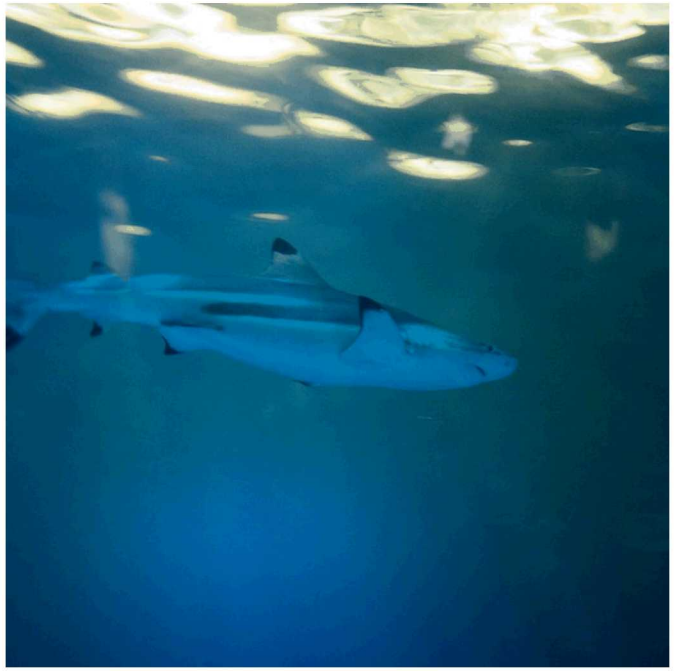
AZUL

```
FACTORY OPTIONS []
[4, 4]
Selected Street [1, 0] 4
Finding Tile Location 1 2
Finding Tile Location 2 2
Factory [[], [], [], [], []]
Street []
Player 0
Live Board:
        1 []
        2 [4, 4]
        3 []
        4 []
        5 []
        Penalties []
        Scoring Board:
        1 [1, 1, 1, 1, 0]
        2 [1, 1, 1, 1, 1]
        3 [0, 0, 0, 1, 0]
        4 [1, 0, 0, 0, 0]
        5 [0, 0, 0, 0, 1]
Player 1
Live Board:
        1 []
        2 []
        3 []
        4 []
        5 []
        Penalties []
        Scoring Board:
        1 [0, 1, 1, 1, 1]
        2 [1, 0, 1, 1, 0]
        3 [0, 0, 1, 0, 0]
        4 [0, 0, 0, 0, 0]
        5 [0, 0, 1, 0, 0]
Player 0 Wins
Game Over
```

Sharks Are Wild

https://gitlab.sandia.gov/gym/sharks-are-wild

```
Discard Pile: 2
Player 1 board: [5] Hand: [1, 8, 0, 2, 8] Score: 0
Player 2 board: [5] Hand: [2, 7, 1, 0, 0] Score: 0
Player 1 Draw from
        0 (shuffled pile)
        1 (discard pile)
```



https://ccsearch.creativecommons.org/photos/e3424645-57ac-4ac9-9369-843d01443620

Rubiks Cube

https://gitlab.sandia.gov/gym/rubiks

https://ccsearch.creativecommons.org/photos/a3512a56-c41e-4d14-a8cf-7fa91a4443af

```
Front Face
W R R
W G G
W R R
Left Face
B R B
B R B
B R B
Right Face
G G G
O O O
G G G
Bottom Face
O W W
B W W
O W W
Top Face
R Y Y
G Y Y
R Y Y
Back Face
Y O O
Y B B
Y O O
```

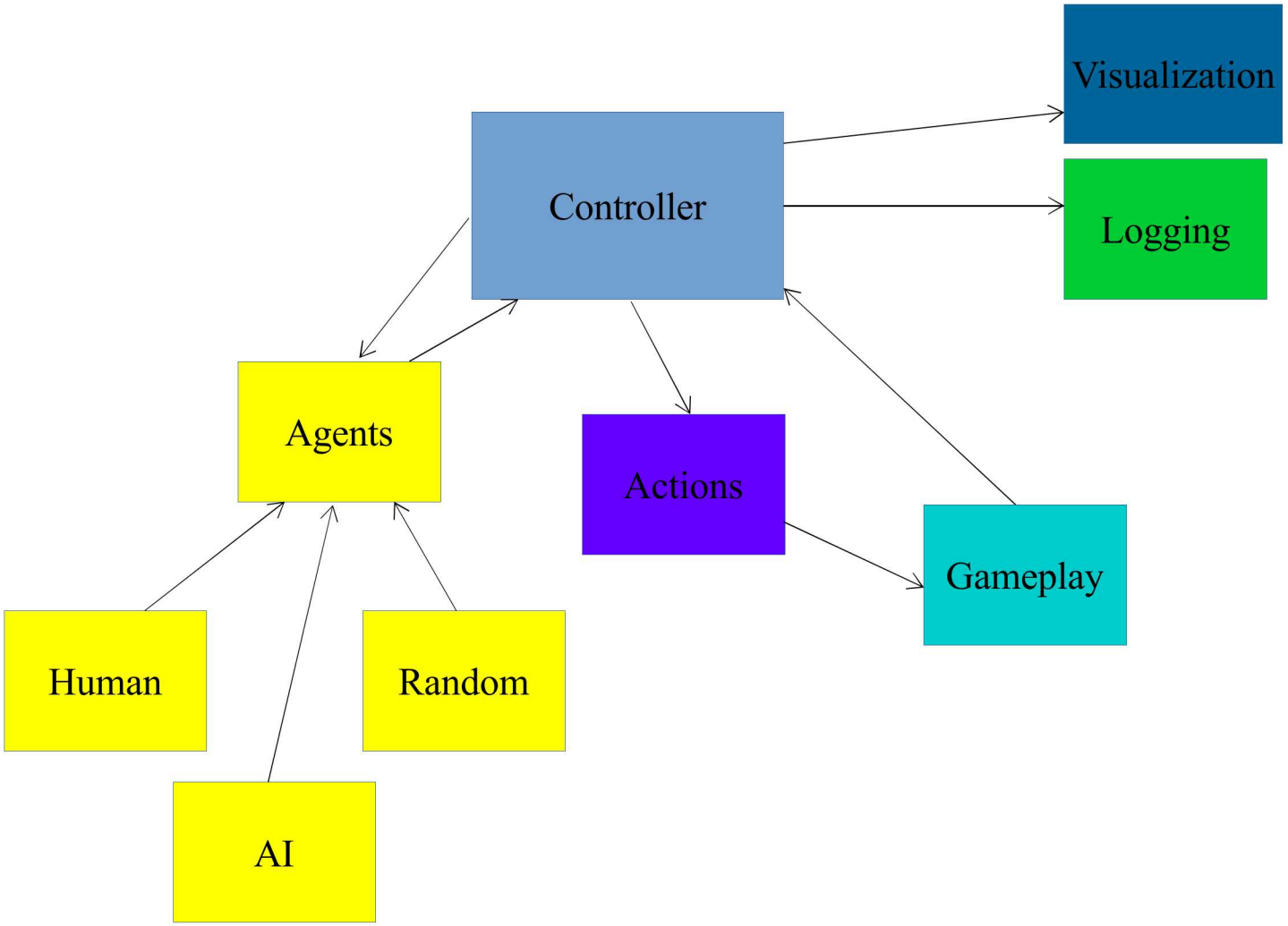
Simulated Intelligent Capture The Key Environment (SICKE)

https://gitlab.sandia.gov/gym/sicke

```
s1 sys/class/devcoredump
s2 proc/19/net/netfilter
s3 sys/bus/vme/drivers
s4 usr/share/doc/login
s5 usr/share/doc/ufw
s6 proc/irq/10
s7 usr/share/doc/python3-configobj
s8 sys/kernel/slab/:at-0000016
s9 proc/395/net
s10 sys/bus/iscsi_flashnode/devices
s11 proc/1036
s12 proc/55
s13 sys/bus/nd/drivers
s18 etc/ldap
s19 usr/share/glib-2.0
s20 usr/share/doc/sed
ubuntu@s7 :/ ls
```

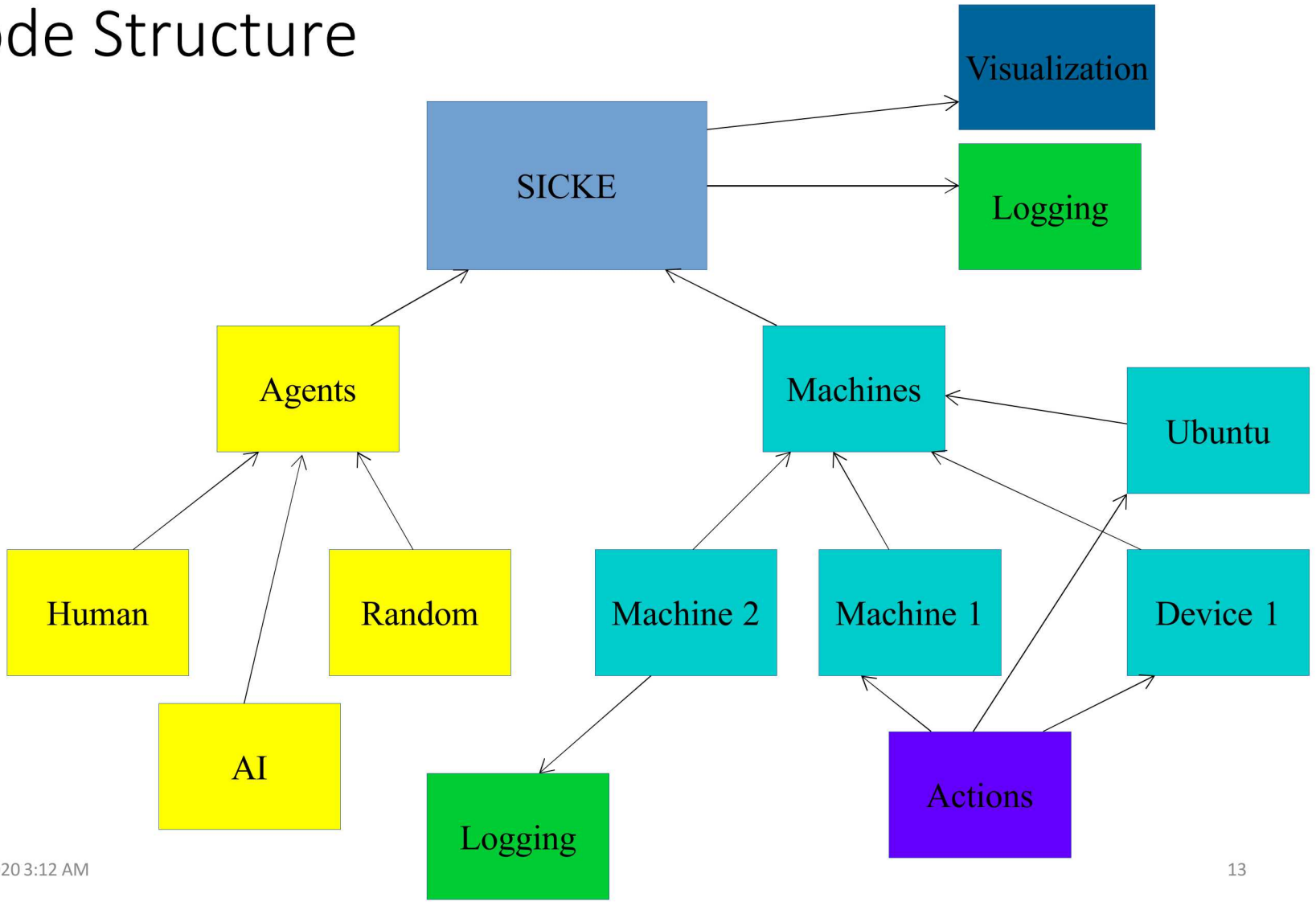
```
ubuntu@s7 :/ ls
bin
boot
dev
etc
lib
lib64
media
mnt
opt
proc
run
sbin
snap
srv
sys
tmp
usr
vagrant
var
Result of action ['bin', 'bo
ubuntu@s7 :/
```

Environment Structure



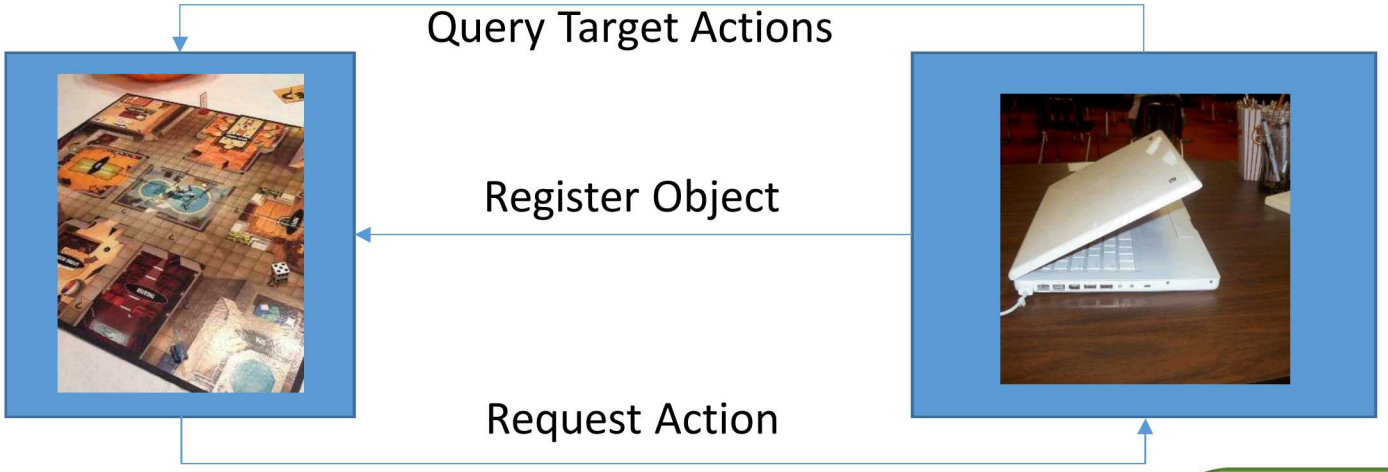
Simulated Intelligent Capture The Key Environment (SICKE)

Code Structure



VahVis

VahVis is a Reinforcement Learning Server, allowing various languages to interact with it as long as they can make HTTP calls. Runs are identified by a unique identifier (UUID). This allows for multiple runs simultaneously managed by UUID.



https://ccsearch.creativecommons.org/photos/eb598513-0f0d-4953-b1b8-8f6e55d77e35

https://ccsearch.creativecommons.org/photos/692a324a-44a8-4282-bd2d-d63291d0eb93

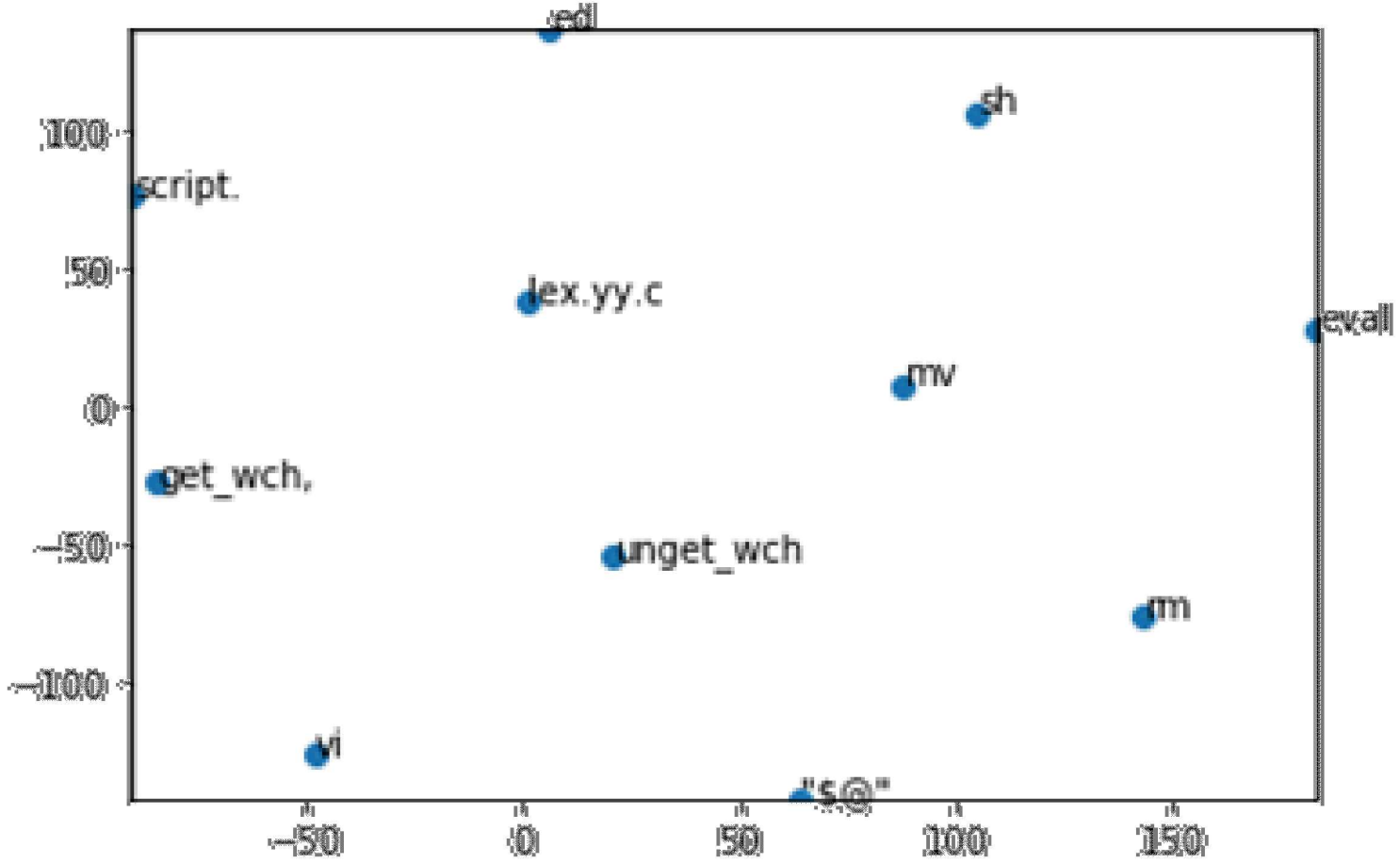
https://gitlab.sandia.gov/vahvis

Benefits:
- GPU Usage (Theano)
- Reuse Existing RL Agents
- Python

Drawbacks:
- Slower than direct on computer
- HTTP Interface necessary

Man2Vec

Ran Word2Vec on a dump of all linux command man pages.



https://gitlab.sandia.gov/tbailey/man2vec

Space

Integrated VahVis into satellite simulation software allowing for RL using curl. This was useful because it didn't require explicitly having C/C++ based reinforcement learning algorithms implemented.

https://ccsearch.creativecommons.org/photos/c723b8c4-72c8-49fa-bf1e-dfaf79d59b8c

Results

There really aren't any. But that's ok.

Building environments for others to experiment can have value.

Experimentation without specific results as a goal is ok as it can prompt new ideas and experiments.

If I hadn't built a few environments beforehand I wouldn't have had an idea for the structure of SICKE.

If I hadn't built VahVis I wouldn't have been able to integrate with satellite software with minimal effort.

Building is learning.

Preguntas?

tbailey@sandia.gov