

How do researchers
keep up?

Set up your personal cluster in the cloud.

Press button to tweak your models

**Massive Computational Experiments,
Painlessly**
(STATS 285)

Stats 285 @ Stanford

cj parrun experiment sherlock

Time: Monday 3:00 - 4:20
Place: Thornt110
Website: stats285.github.io

- Computational Science/Computation-enabled discovery is becoming mainstream!
- PhD Students are expected to conduct **1 million CPU-hour** of computation
- Personal Laptops → Shared Clusters → Personal Clusters

Research adaptation in modern age

- Computing Change demands adaptation:
 - **Pose** bold research **hypotheses** to settle computationally
 - **Design massive computing experiments**
 - **Adopt** painless computing **frameworks**
 - **Raise money** to pay for cloud-based computing
 - *Push a button*

However, handling the
computation interface
is painful!

Rethinking computational experiments

In our telling, a computational experiment involves:

- 1 **Precise Specification** (define metric and parameters)
- 2 **Execution and management** of all the jobs
- 3 **Harvesting** of all the data generated by all the jobs
- 4 **Analysis** of the data
- 5 **Reporting** of results.

Today we add to this list: **Building** of compute clusters

The painless computing paradigm should seamlessly integrate and automate all these tasks

Frameworks to automate computation from Stats285

Many open-source frameworks offer automation
painless (push-button) massive computing

- Building Cloud Clusters
 - **ElastiCluster** (Riccardo Murri)
 - StarCluster (MIT)
- Experiment Management Systems (Laptop → Cluster)
 - **CJ** (Yours Truly)
 - CodaLab (Percy Liang)
 - PyWren(serverless) (Eric Jonas)
- Machine Learning and Statistics
 - **PyTorch**, Tensorflow, CNTK, Theano, Keras
 - Spark, Dask

ClusterJob



[Documentation](#) [People](#) [Support](#)

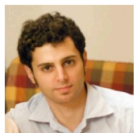
[Sign in](#)

[Sign up](#)

Reproducible
High-throughput
Computational Research

Sign up to access your computations
anytime, anywhere.

[Sign up for Clusterjob](#)

The hero banner features a background image of a large, ornate building with a tiled roof and a central gable containing a mural. The text is overlaid on the left side, and a green call-to-action button is positioned on the right.

“This is how it [computation] should be done.” – V. Morgenshtern



“Your software has made my life much easier.” – C. Chang

ClusterJob's vision

- *push a button, fire and forget*
- *harvest, analyze and publish discovery*



Look into CJ

Open source project:

`https://github.com/monajemi/clusterjob`

The screenshot shows the GitHub repository page for `monajemi/clusterjob`. At the top, there is a navigation bar with links for Features, Business, Explore, Marketplace, Pricing, This repository, Search, Sign in, and Sign up. Below the navigation bar, the repository name `monajemi/clusterjob` is displayed, along with statistics: 1 Watch, 3 Stars, and 1 Fork. A navigation menu includes Code, Issues (8), Pull requests (0), Projects (0), Wiki, and Insights. The repository description is "ClusterJob: An automated system for painless and reproducible massive computational experiments" with a link to `http://clusterjob.org`. Below the description, there are statistics: 348 commits, 4 branches, 4 releases, 1 contributor, and BSD-3-Clause license. A section for the latest commit shows a fix by monajemi, with a commit hash of 937757c, committed 9 days ago. Below this, there are two more commits: one for 'example' fixed 9 days ago, and one for 'src' with a message '&& added instaedof :' committed 10 days ago. At the bottom, the 'Initial commit' is shown, committed by monajemi on Feb 26, 2015, with a commit hash of ab21b80.



ElastiCluster: abstract away infrastructure layer

What is ElastiCluster

ElastiCluster provides a **command line tool** and a Python API to **create, set up and resize** computing clusters hosted on IaaS cloud infrastructures.

Main function is to get a compute cluster up and running with a single command.

Build your own cluster

- Setup Google Cloud Billing (*Yes, they will charge you!*)

- 1 gmail username
- 2 project_id
- 3 client_id
- 4 client_secret

- Install Docker on your machine

- removes the pain and ir-reproducibility of software install

- Create your cluster using dockerized ElastiCluster

```
docker image pull stats285/elasticcluster-gpu
```

```
docker run -it stats285/elasticcluster-gpu
```

```
elasticcluster start gce
```



Push button to run experiment!

Action items for conducting experiments:

```
1) elasticcluster start gce
```

```
2) cj parrun train.py gce -alloc "--gres=gpu:1"
```

See Stats285 HW2

Massive Computational Experiments, Painlessly (STATS 285)

Stanford University, Fall 2017

Assignment 02

In this assignment, we will conduct a collaborative project testing certain theoretical hypotheses in Deep Learning. In particular, each of you will build **your own personal SLURM cluster** on Google Compute Engine (GCE) using [elasticcluster](#) and then run massive computational experiments using [clusterjob](#). We then collect and analyse all the results you will generate and document our observations. Please follow the following step to setup your cluster and run experiments. This documents only contains the detail of setting up your cluster and testing that it works properly with

A summary

- Small computations → Massive computations
- Interactive model (copy-paste) → experiment management systems (EMS)
- Personal Laptops → Shared Clusters → Personal Clusters
- Expansion by factors of 1000's in immediate computing capacity
- Rise of frameworks takes away pain of massive computing
- Deep Learning is now a technology (Thanks to frameworks)
- Lots of opportunities for semi-empirical/semi-theoretical Deep Learning studies

Computation is so easy now!

Is this the end of story?

Questions not addressed: Lack of computational resource!

- Research progress depending on the amount of computational resource at hand!
- More resource you have, more computational tweak you can try.

We solved only half the problem!

- Our solution turns computational resource into a problem of monetary budget.
- But how do people get funded for their AI research?

Two solutions

- Centralized way
 - Academic infrastructure
 - Research grant
 - Free credits on cloud
- Decentralized way
 - Community sharing?
 - Incentives?

A community for AI models

Two research approach for deep learning

- Rich man's approach:
 - Train all candidate models from scratch
 - Possibly on several large datasets
 - Try all tweaks!
 - Save all the intermediate results

Two research approach for deep learning

- Poor man's approach:
 - Take pre-trained models
 - Transfer learning on small dataset
 - Analyze the intermediate results from other people (Rich men)

A sharing, friendly community would help the ecosystem to grow happily!

- Rich man publishes research result with a variety of solid computational results.
- Poor man can look into the gold mining of results and observe interesting phenomenon based on the unique angle, without much computation.

What kind of community do we need?

- In the software age, code sharing is done by Github, Gitlab, etc.
- In the AI age, code sharing is not enough!

Why GitHub is not **GitHub** for **AI** ?

Model \neq Code

Discoverability

Monetary Incentive

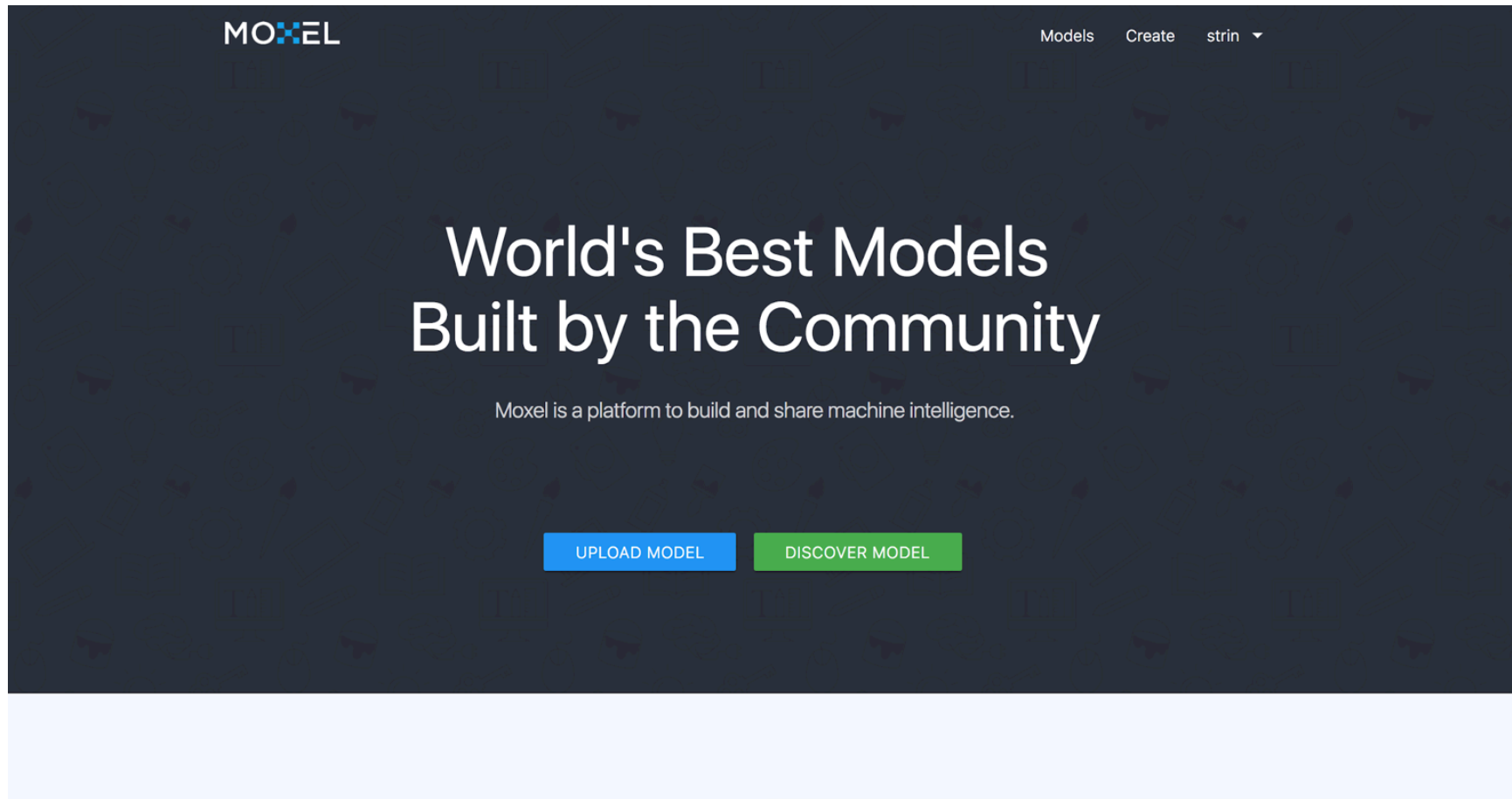
Deep Learning Model \neq Code

- Deep Learning Model = code + environment + dataset + trained parameters + output interface
- Deep Learning Model history consists of the entire training process snapshot

In general, we think of model as a trainable function that eats input (data) and spits output.

Model as a trainable
program.
(Software 2.0)

Moxel: Github for AI models



Tensorflow Object Detection Model

▲ 18

SHARE

LIVE

LATEST

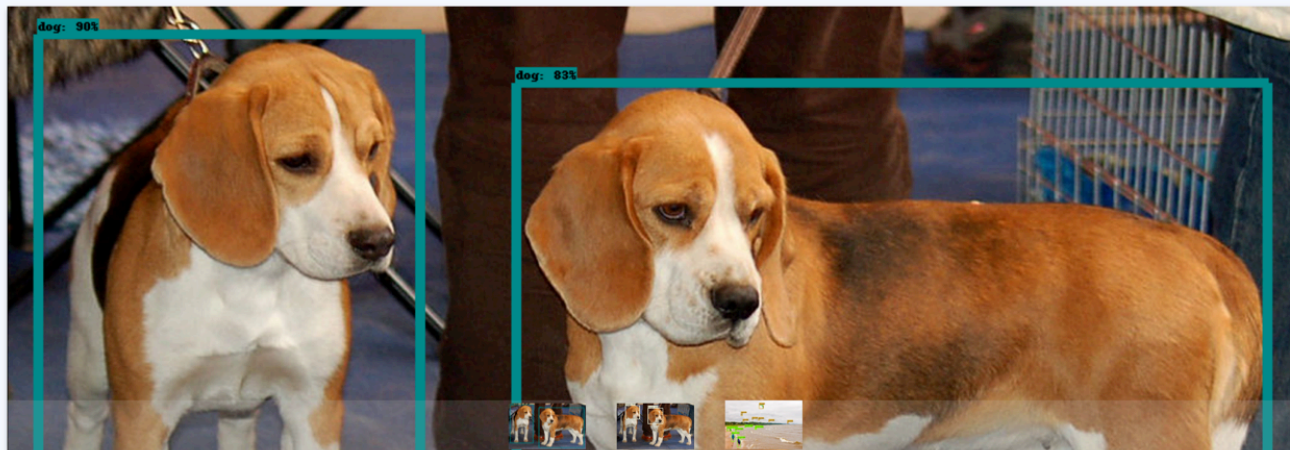
This is a tensorflow object detection model.

tensorflow

computer vision

GITHUB

ARXIV



DEMO

API

INPUT TYPE

OUTPUT TYPE



<http://beta.moxel.ai/models/strin/colorization/latest>

Image Colorization Model

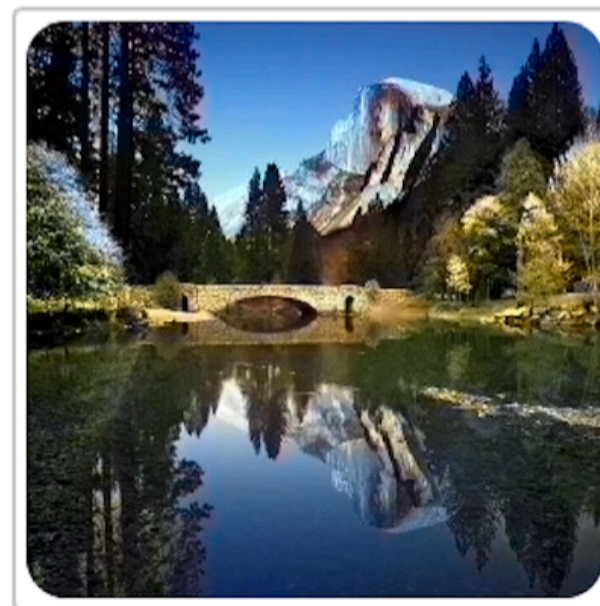
Model Input

img_in image



Model Output

img_out image



<http://beta.moxel.ai/models/moxel/neural-storyteller/latest>

Neural Networks Tell Little Stories about Images

Model Input

image image



Model Output

passage str

Everyone was at the beach , someone taking it slow and steady . I have never seen the visions of the ocean before , as I walked back out of the ocean to greet him . The horse had changed so much , and I felt like I was leaving her .

<http://beta.moxel.ai/models/strin/bi-att-flow/latest>

Bi-directional Attention Flow for Question-Answering

Model Input

paragraph **str**

Artificial intelligence is set to revolutionize every sector, and will likely eliminate whole categories of jobs in the next few years, said Andrew Ng, a Stanford adjunct professor and former Baidu scientist, at the WSJ

question **str**

Who is Andrew Ng?

Model Output

answer **str**

Stanford adjunct professor and former Baidu scientist

A Marketplace, A Cloud Platform.

Researchers
Deploy & Publish Models

Model A

Model X

Model B

Discover & Use Models

Moxel
Cloud Infrastructure: Serving,
Scaling, Security.

Upload your model and it goes live!

Moxel as a prediction API wrapper

```
1 import moxel
2
3 model = moxel.Model('google/tensorflow-object-detection')
4 result = model.predict(image)
```

Upload your model to help the world discover your work

<http://beta.moxel.ai>

Uploading models is currently invite-only.

The access code to unlock is:

backprop

Tutorial

- <http://docs.moxel.ai/tutorials/question-answering/>

Models and their empirical results should be the final products of empirical researches.

Papers should only be summaries of the work!

Moxel as the online Arxiv for models! The NIPS review process is replaced by the voting scheme in Moxel community.

Hot Models of the Week

October 26 - Issue #1

A weekly newsletter on AI models curated by Moxel.

Check out these awesome models built by the community! If you like a model, please consider sharing it on Twitter, Facebook, etc. We really appreciate the support.

Models

SqueezeNet in Pytorch

jimfan/squeezenet/latest

SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size

28 Stars *93 Views* *96 Runs*

Generating Compositional Color Descriptions

futurulus/color-describer/latest

This is the model from the paper Learning to Generate Compositional Color Descriptions (Monroe, Goodman & Potts, 2016). It takes a color as input and produces a description based on data from the xkcd Color Survey assembled by McMahan & Stone (2015). Try running it more than once with the same color!

3 Stars *161 Views* *398 Runs*

A Friendly Chatbot

awni/chatbot-v0/latest

A neural sequence to sequence chatbot trained on the Cornell Movie Dialogue Corpus

4 Stars *24 Views* *30 Runs*

Adversarial attacker to ImageNet classifiers

zhangyuc/attacker/latest

An ensemble attacker to fool Inception V3 ImageNet classifiers. Try generate adversarial examples here, and send it to Inception V3 at: beta.moxel.ai/models/strin/inception-v3

2 Stars *46 Views* *44 Runs*

Neural Networks Tell Romantic Stories From Images

strin/neural-storyteller-gpu/latest

neural-storyteller is a recurrent neural network that tells romantic stories based on images. It uses a VGG network to encode image and generate captions from COCO dataset. Then it aligns the COCO captions with a romantic corpus. Finally, it outputs a sentence using a decoder trained on the romantic corpus. Try uploading different images and see what happens!

26 Stars *554 Views* *493 Runs*

How to Grow the Community?

Gather A Cohort of **High-Quality** Models.

Make Model Deployment Frictionless

```
~/awesome-model: warp create -f dummy.yml
> Model awesome-ai:latest
> Pushing code...
> Commit 81ed8093ec9560b30f221c6d067f00ce53e211b7
> Uploading weight files...
 9.88 MiB / 9.88 MiB [████████████████████████████████████████] 100.00% 15s
Done!
~/awesome-model: warp deploy awesome-ai:latest
successfully deployed awesome-ai:latest
~/awesome-model: █
```

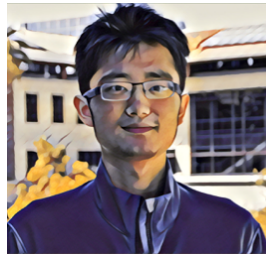
We want to have a community of AI researchers.
How do we encourage researchers to upload their models?

Need incentives!

Incentives

- Turn AI model into a API
- Users pay for API Calls
- Encourage model sharing

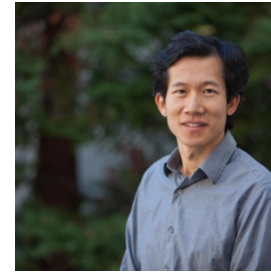
Joint work with



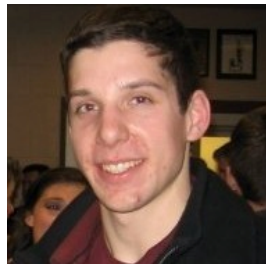
Tim Shi



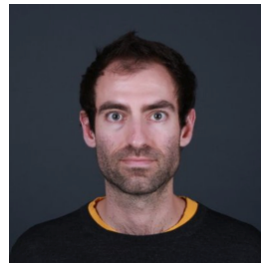
Jim Fan



Percy Liang



Chris Lengerich



Awni Hannun