NSML: A Machine Learning Platform That Enables You Focus on Your Models.

ML-Sys WS 2017 @ NIPS

Nako Sung, Minkyu Kim, Hyunwoo Jo, Youngil Yang,
Jinwoong Kim, Leonard Lausen, Youngkwan Kim,
Gayoung Lee, Donghyun Kwak, Jung-Woo Ha, and Sunghun Kim

CLOVA AI Research (CLAIR), NAVER | LINE, Search Solution, NAVER Webtoon, HKUST

What is NSML?

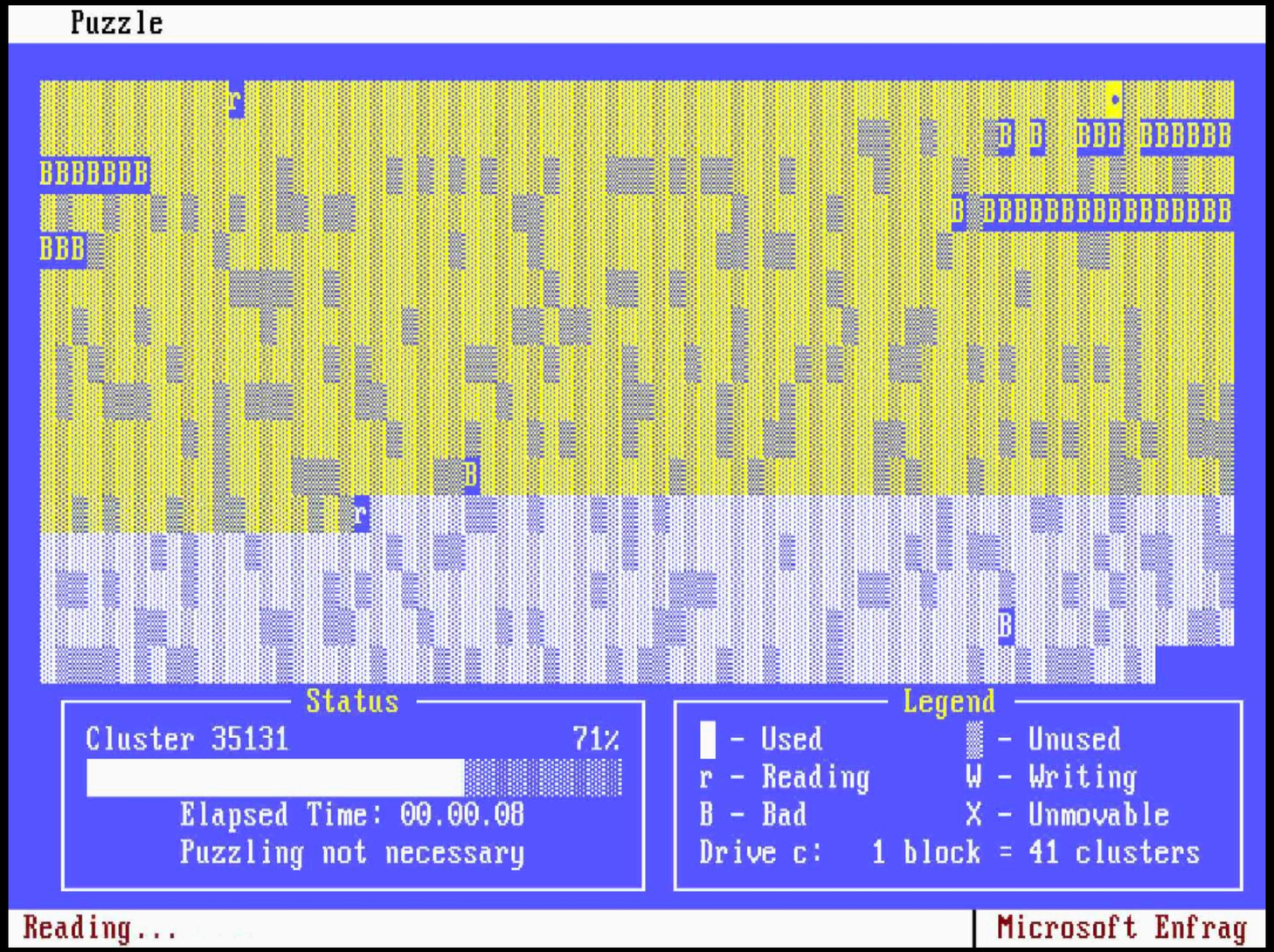
- A machine learning platform that enables you focus on your models
- Two options: on-premise / PaaS

The #1 programmer excuse for legitimately slacking off - 2017 version

https://xkcd.com/303/

NAVER

LINE

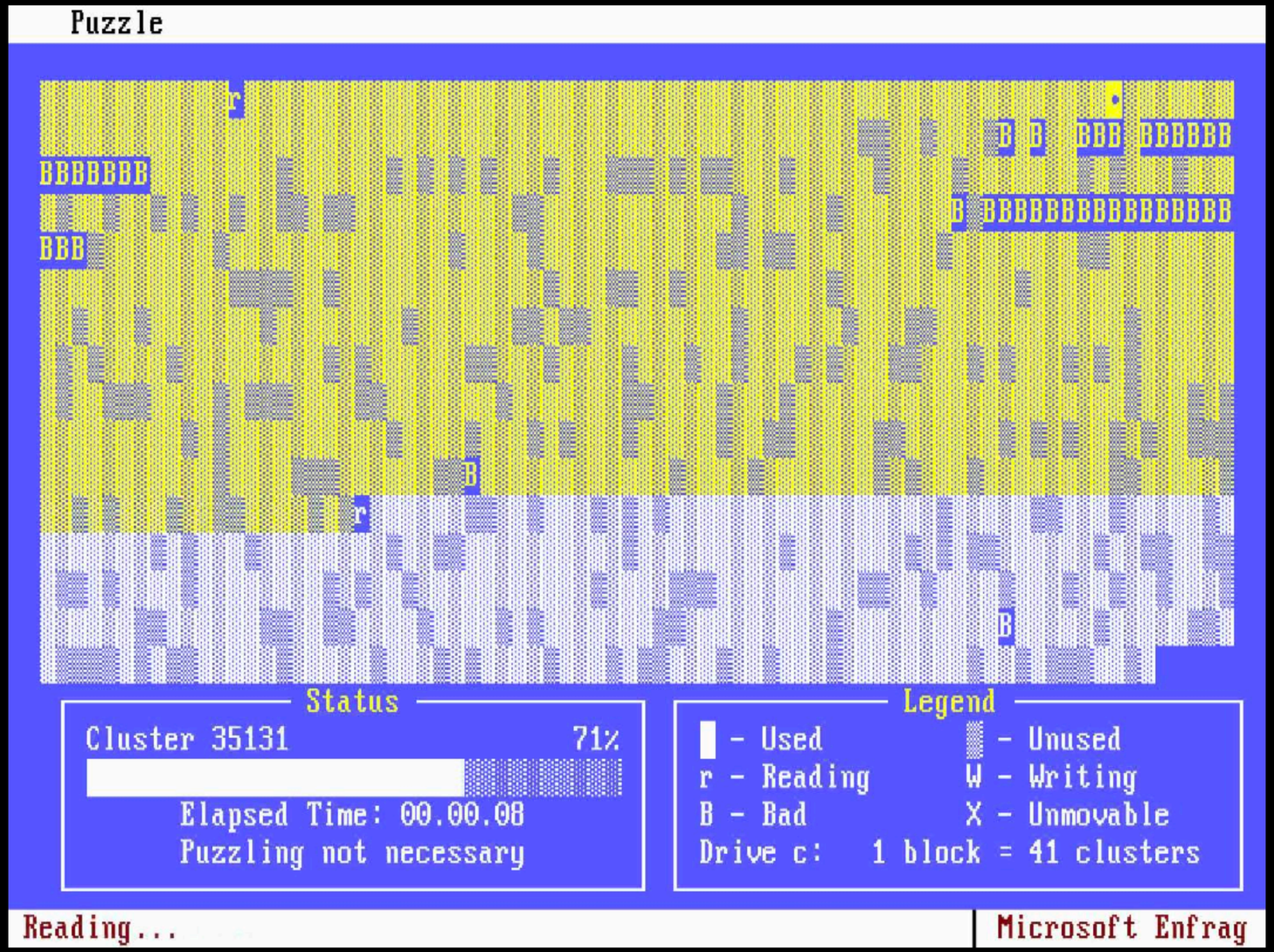


https://www.youtube.com/watch?v=lxZyxxHOw3Y

NAVER

LINE

Wasted Time



https://www.youtube.com/watch?v=lxZyxxHOw3Y

NAVER

LINE

https://www.formula1.com/en/latest/features/2017/2/F1-cars-of-2017.html

Importance of Fast Machines (Multiple Servers and GPUs)

https://www.formula1.com/en/latest/features/2017/2/F1-cars-of-2017.html

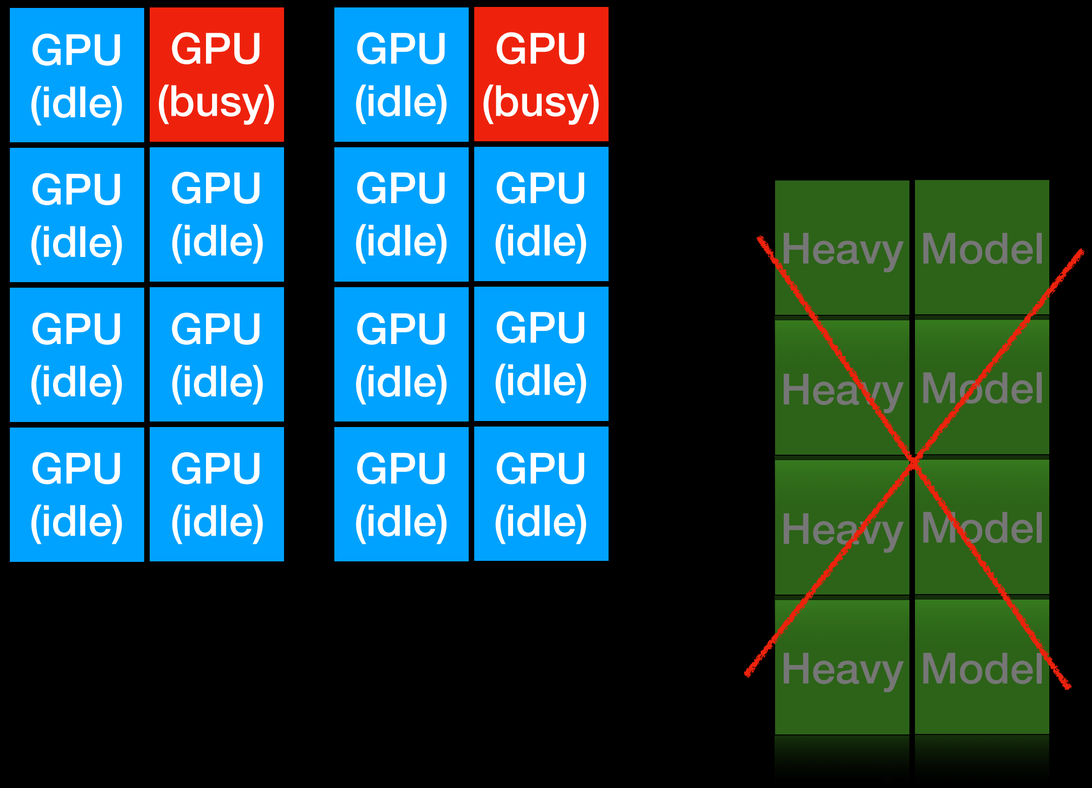
https://www.sportskeeda.com/f1/what-happens-during-f1-pit-stop

ML Research Challenges: Incidental Tasks

https://www.sportskeeda.com/f1/what-happens-during-f1-pit-stop

NAVER

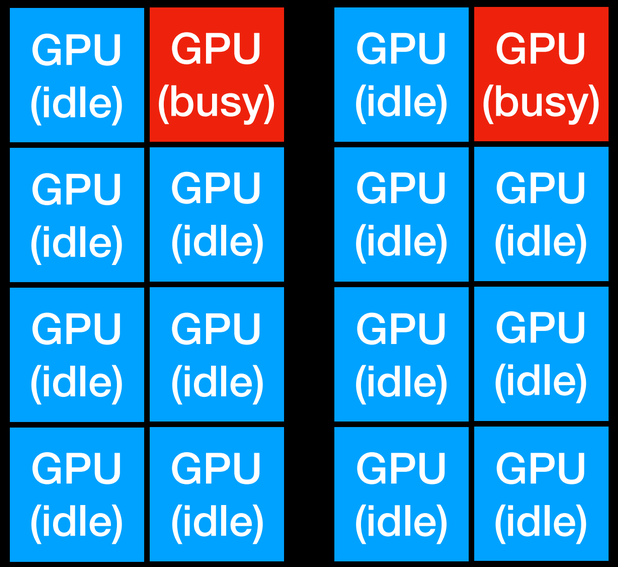
LINE


GPU (idle)
GPU (busy)
GPU (idle)
GPU (busy)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
GPU (idle)
Heavy Model
Heavy Model
Heavy Model
Heavy Model

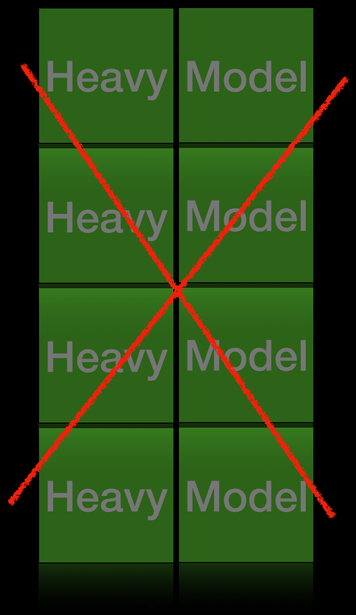
NAVER

LINE

ML Research Challenges: Resource Scheduling and Utilization

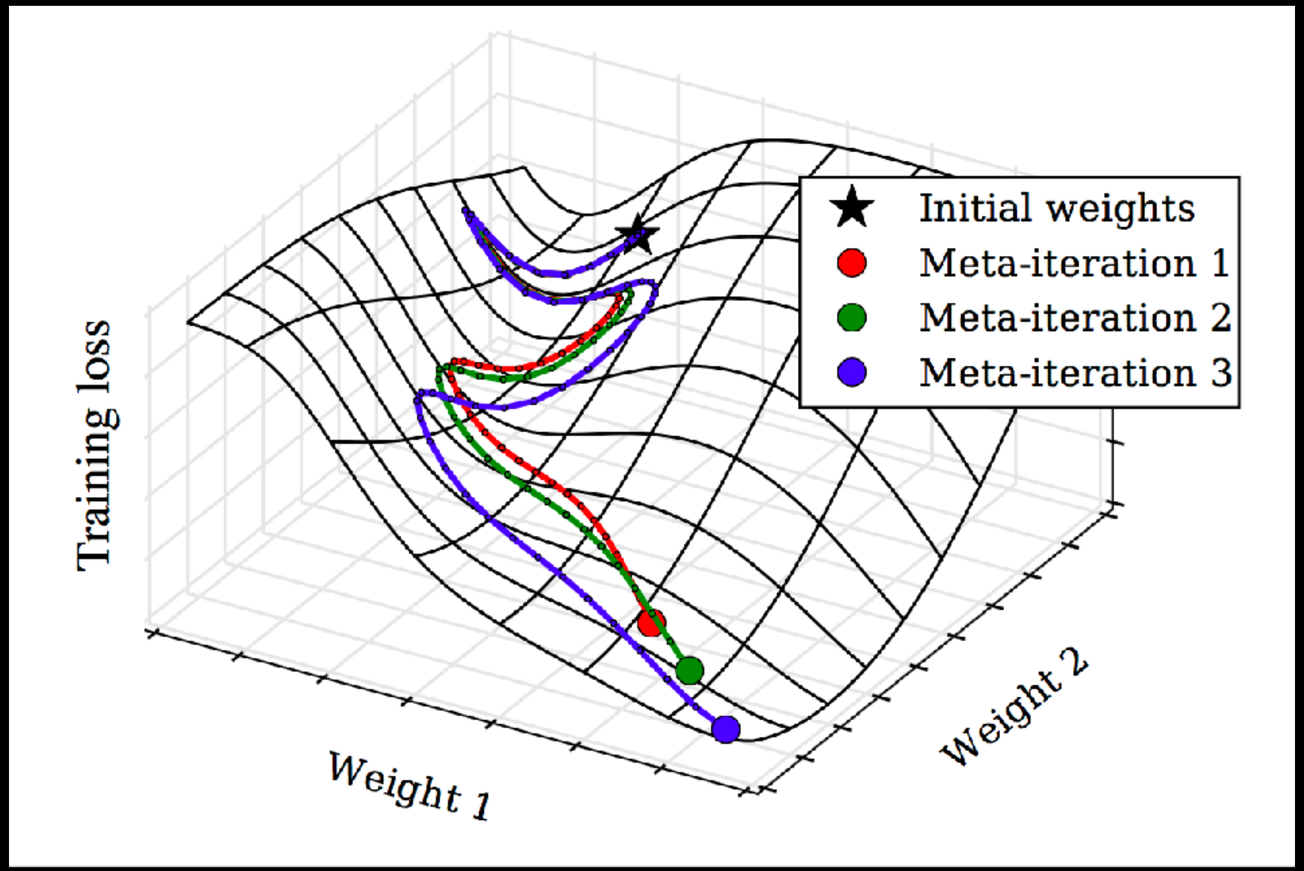


14 GPUs available but only 7 GPUs can be used in a single machine.



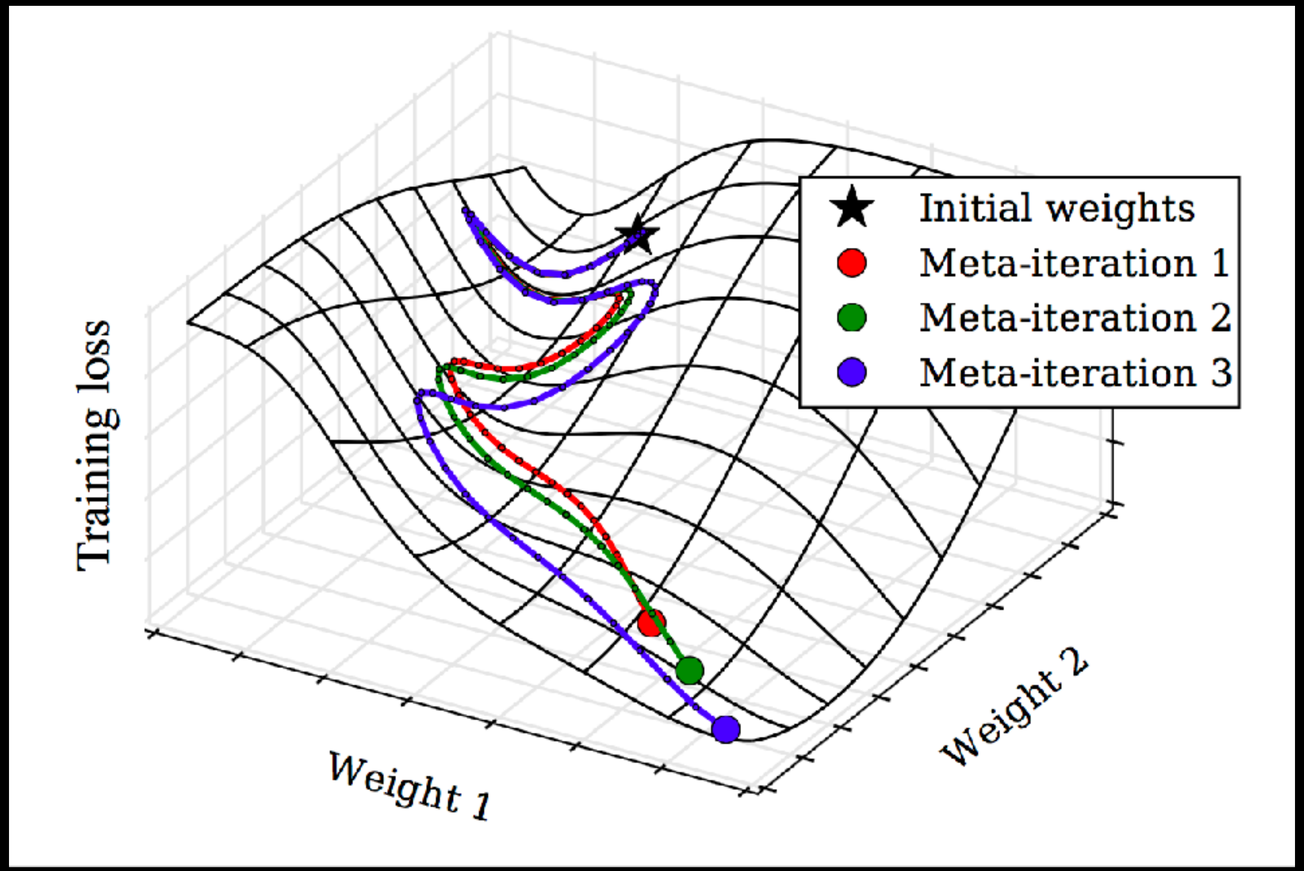
NAVER

LINE



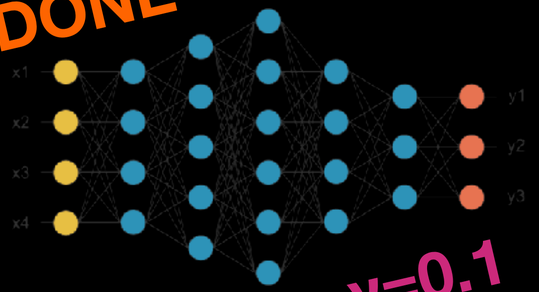
https://livingthing.danmackinlay.name/automl.html

ML Research Challenges: Hyperparameter Tuning



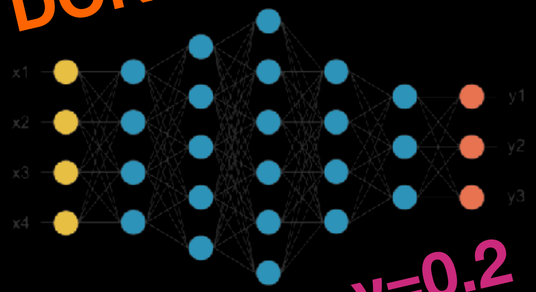
https://livingthing.danmackinlay.name/automl.html

DONE



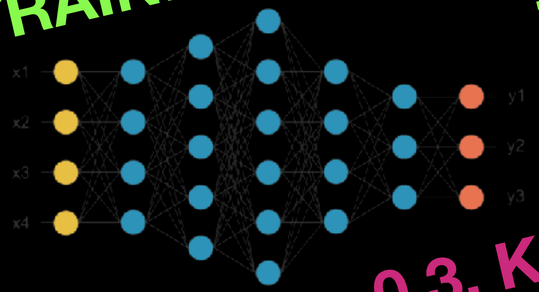
γ=0.1

DONE



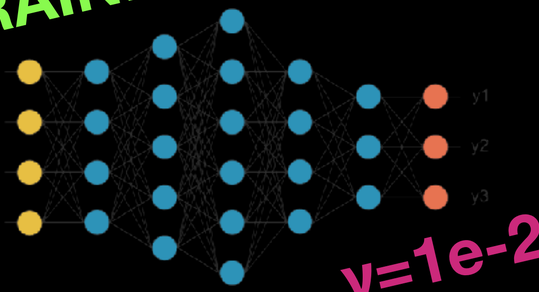
γ=0.2

TRAINING

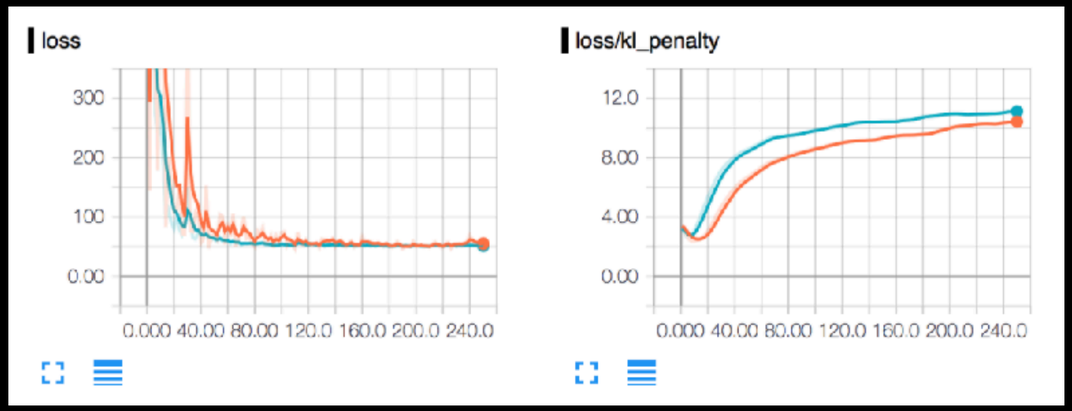


γ=0.3, K=1

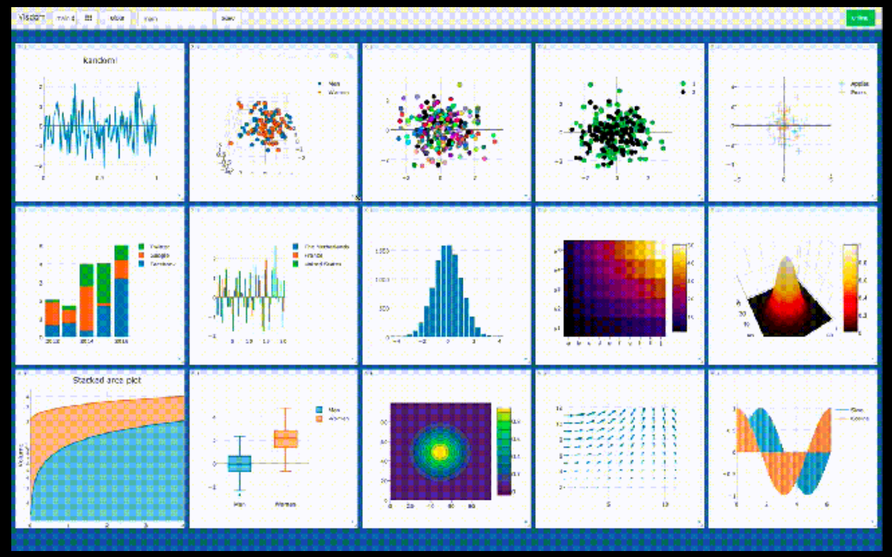
TRAINING



γ=1e-2



Tensor board

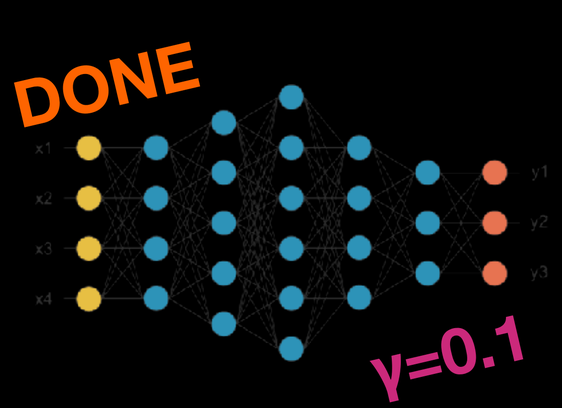


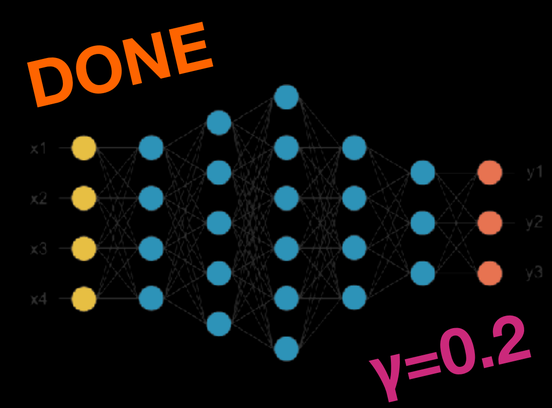
Visdom

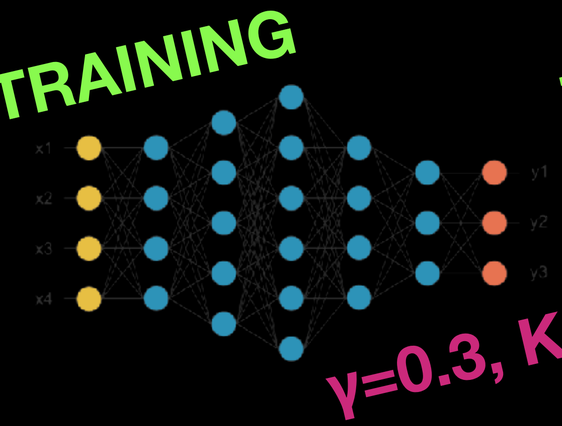
NAVER

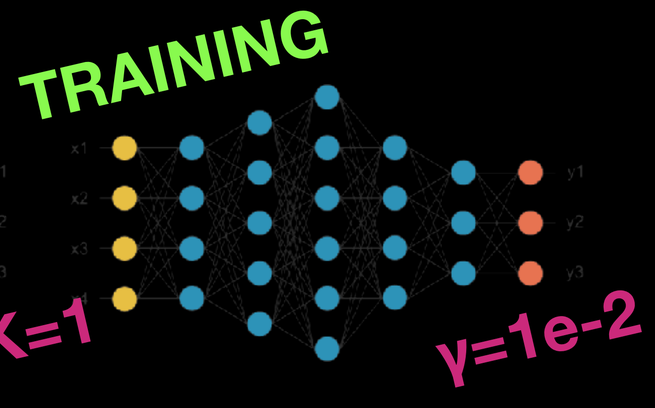
LINE

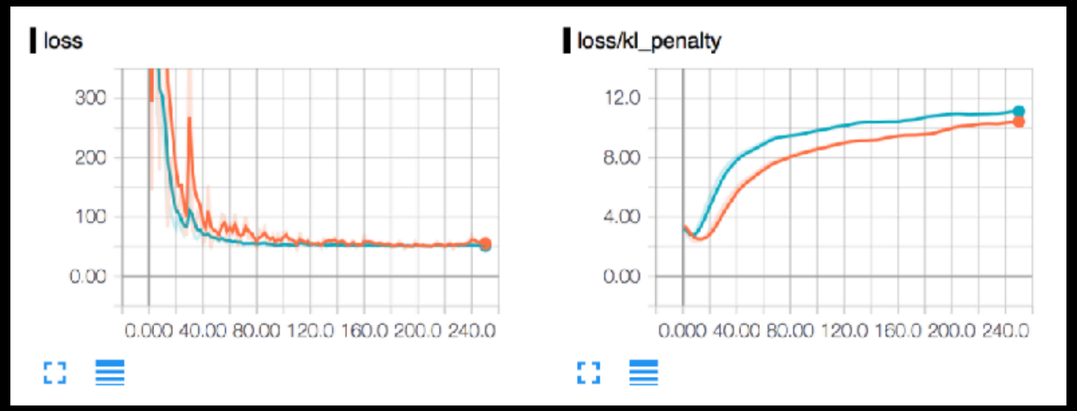
ML Research Challenges: Multiple Experiments



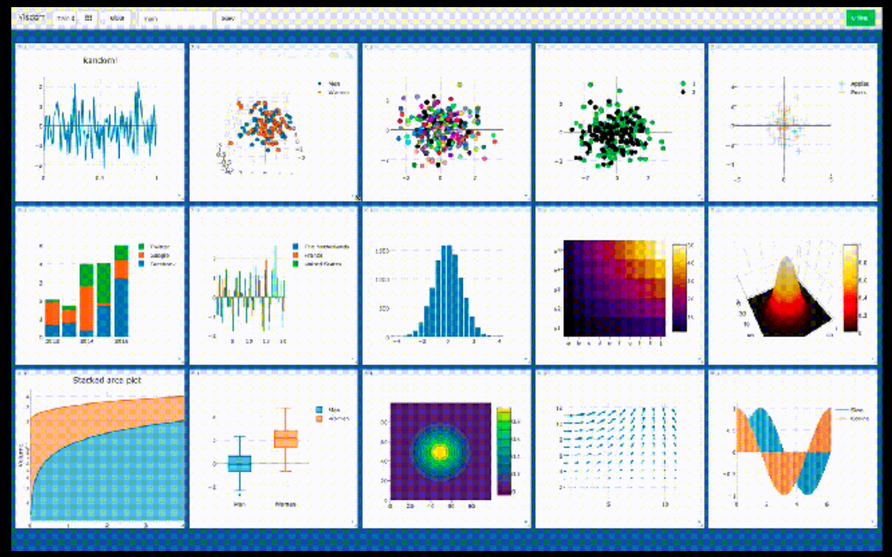








Tensor board



Visdom

NAVER

LINE

https://www.linkedin.com/pulse/protecting-workers-who-work-alone-sandie-baillargeon

ML Research Challenges: Isolated Researchers

https://www.linkedin.com/pulse/protecting-workers-who-work-alone-sandie-baillargeon

NAVER LINE

Challenges

- Slack
- Incidental Tasks
- Inefficient resource utilization
- Naive hyperparameter tuning
- Painful keeping track of multiple sessions
- Isolated researchers

Requirements of ML Platforms

- Resource Management
  - Better computational resource management
- Data Management
  - Post datasets once and reuse them for multiple models
  - Share datasets with others
- Serverless Configuration
  - No framework / library lock-in
  - Easy and lightweight task submission

Requirements of ML Platforms

- Experiment Management and Visualization
  - Parallel runs with different jobs priorities
  - Automatic visualization and summarization of learning progress
- Leaderboard
  - Leaderboard for each dataset to compare models and hyper parameters
- AutoML
  - Experiment performance prediction based on previously run experiments.
  - Automatic hyper parameter optimization based on the performance predictions.

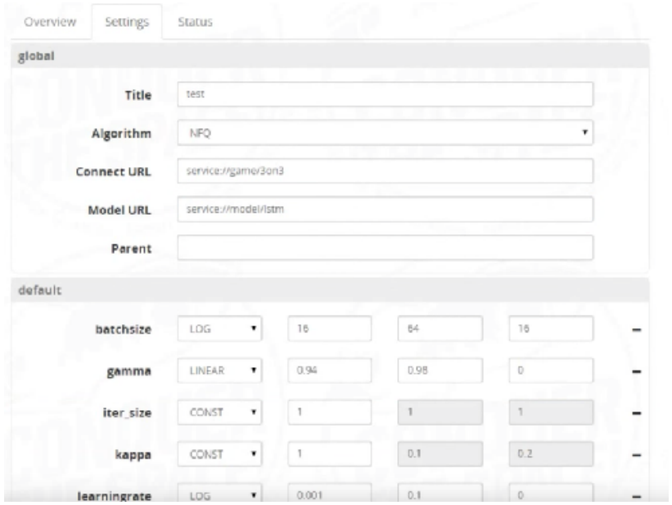
NAVER LINE

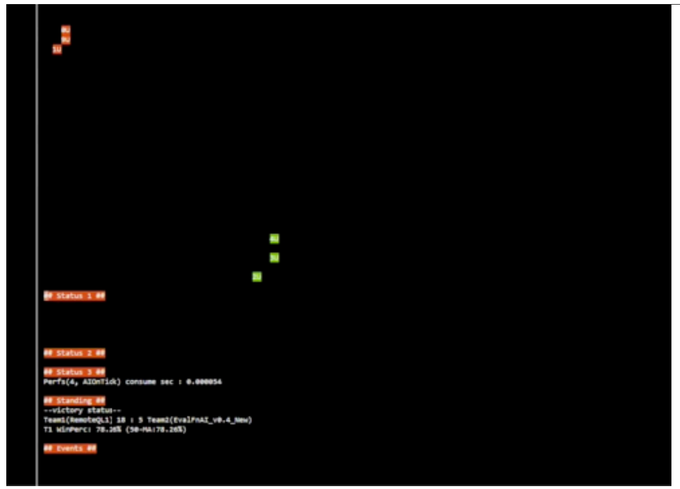
Limitations of Previous Solutions

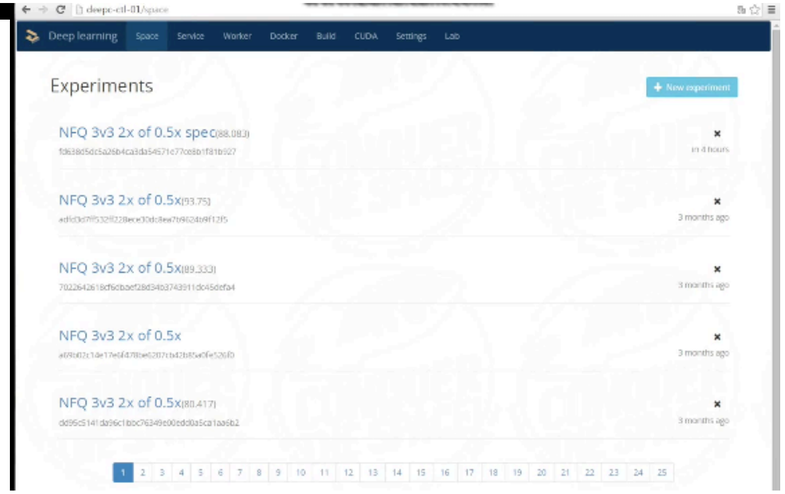
- Vendor lock-in (Cloud service)
- Inefficient model experiments
- Inconsistent research environments
- Still hard to keep track of experiments

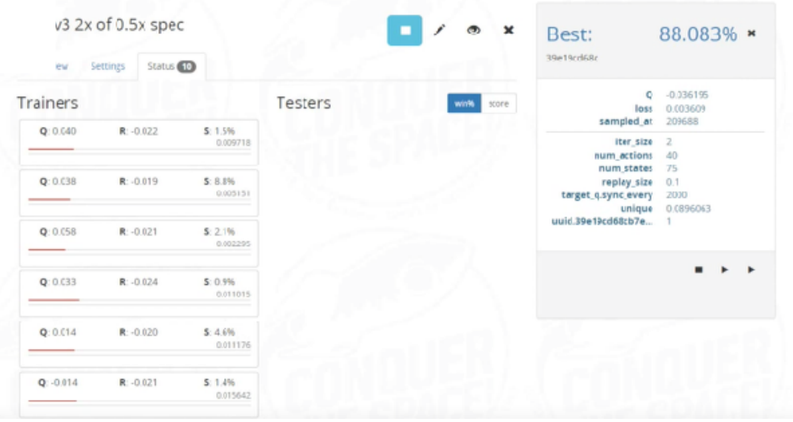
This work was done for NCSoft and was presented at Nvidia GTC Korea 2015.

MINI LEAGUE OF LEGENDS









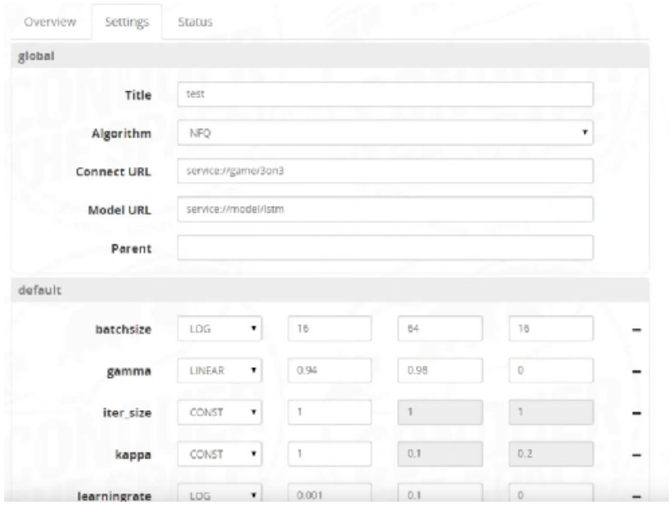
NAVER

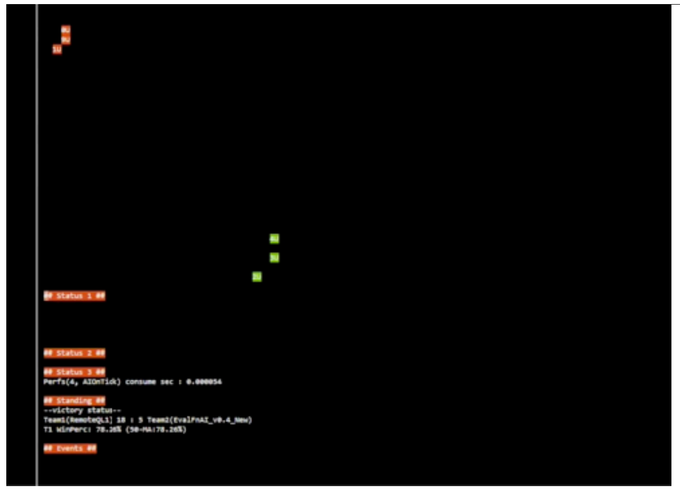
LINE

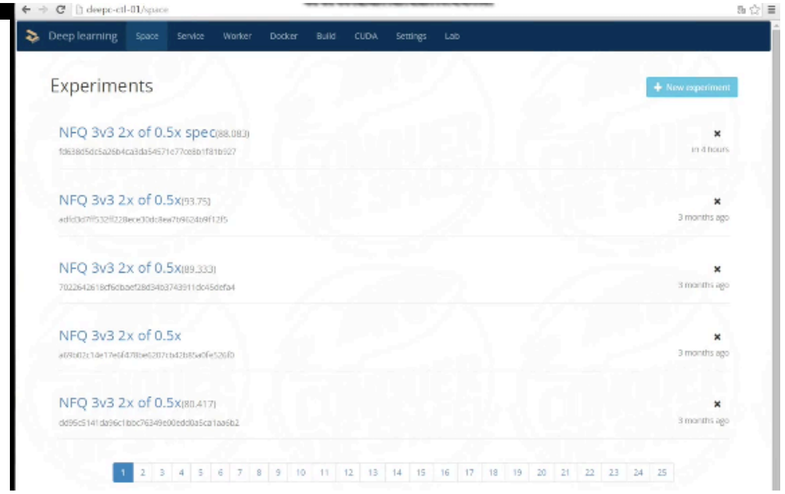
This work was done for NCSoft and was presented at Nvidia GTC Korea 2015.

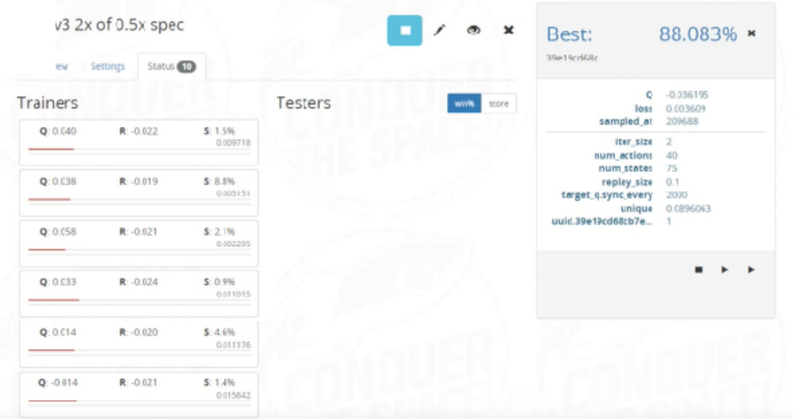
My Previous Work in Early 2015

MINI LEAGUE OF LEGENDS





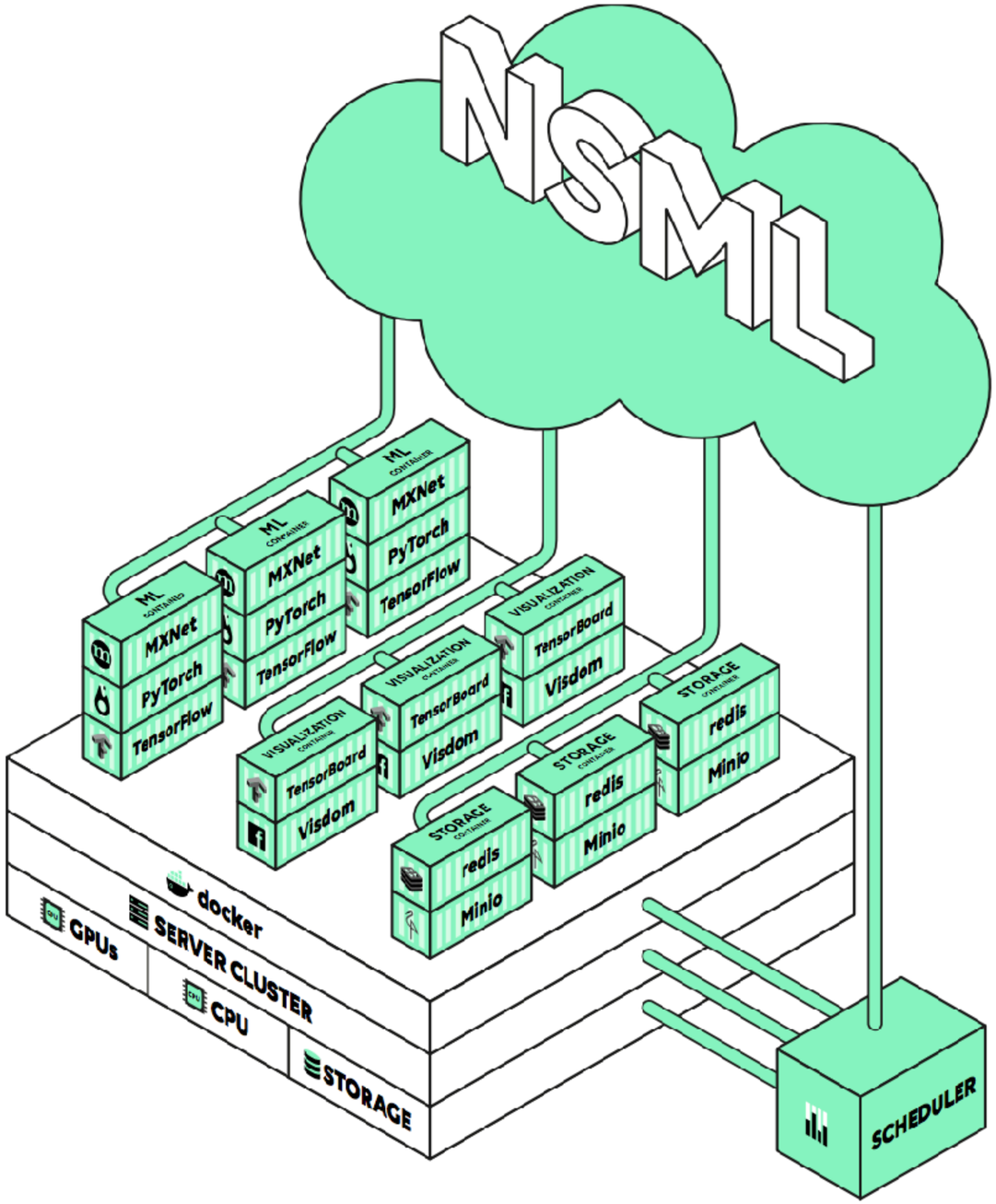




NAVER

LINE

Fork me on GitHub



NAVER

LINE

URI

{Dataset} / {User id} / {Session id} / {Model id}

- Every dataset, session and model have uniform resource identifier.

CIFAR_10	CIFAR 10 dataset
CIFAR_10/researcher_A/24	research_A's 24th session for CIFAR_10
CIFAR_10/researcher_A/24/322	Snapshot from epoch 322

NAVER LINE

Easy One-Liner CLI

Easy One-Liner CLI

Dataset registration

```
/app/examples/09_NMT$ nsml dataset push NMT_EN_KR ./nmt_en_kr
```

Easy One-Liner CLI

Dataset registration

```
/app/examples/09_NMT$ nsml dataset push NMT_EN_KR ./nmt_en_kr
```

Train

```
/app/examples/09_NMT$ nsml run -d NMT_EN_KR
Session clair/NMT_EN_KR/1 is running
```

NAVER LINE

Easy One-Liner CLI

Dataset registration

```
/app/examples/09_NMT$ nsml dataset push NMT_EN_KR ./nmt_en_kr
```

Train

```
/app/examples/09_NMT$ nsml run -d NMT_EN_KR
Session clair/NMT_EN_KR/1 is running
```

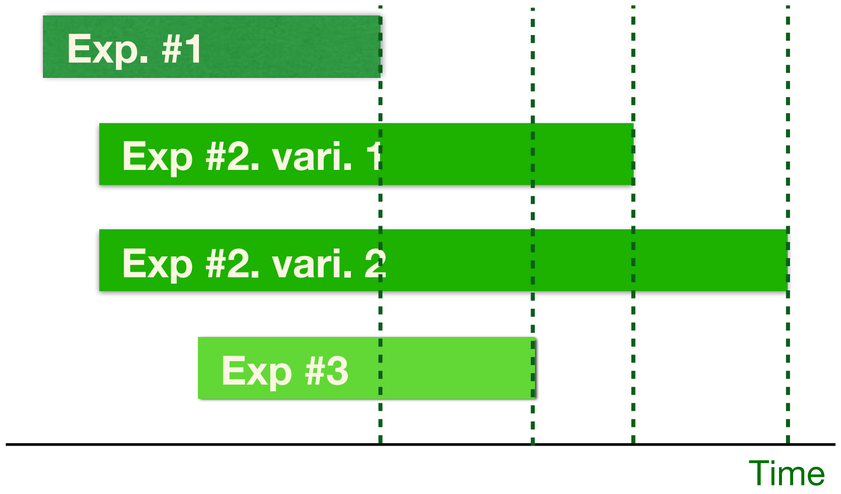
Serve

```
/app/examples$ echo Hello | nsml infer clair/NMT_EN_KR/1/12
안 녕 하 세 요
```

NAVER LINE

Parallel Experiments to Kill Slack

Distributed responses

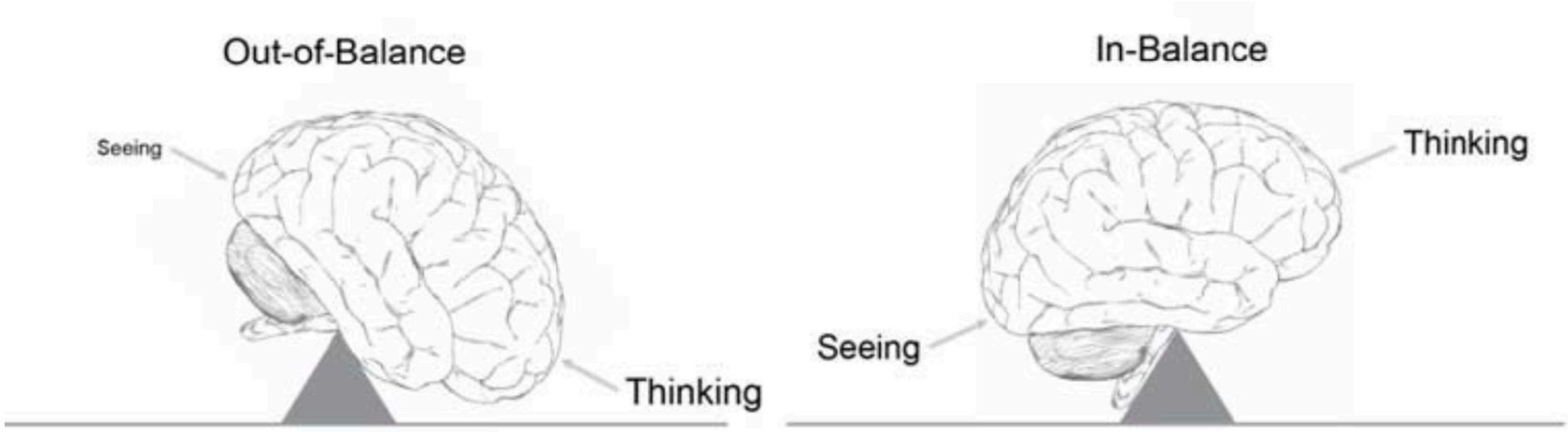



```
/app/examples/02_mnist$ nsml run main.py -- --lr 0.1
Session KR18284/None/12 is running
/app/examples/02_mnist$ nsml run main.py -- --lr 0.01
Session KR18284/None/13 is running
/app/examples/02_mnist$ nsml run main.py -- --lr 0.001
Session KR18284/None/14 is running
/app/examples/02_mnist$ nsml ps
Name                    Created       Args
--------------------  ------------  ----------------------
KR18284/None/14         just now      main.py --lr 0.001
KR18284/None/13         just now      main.py --lr 0.01
KR18284/None/12         seconds ago   main.py --lr 0.1
```

NAVER LINE

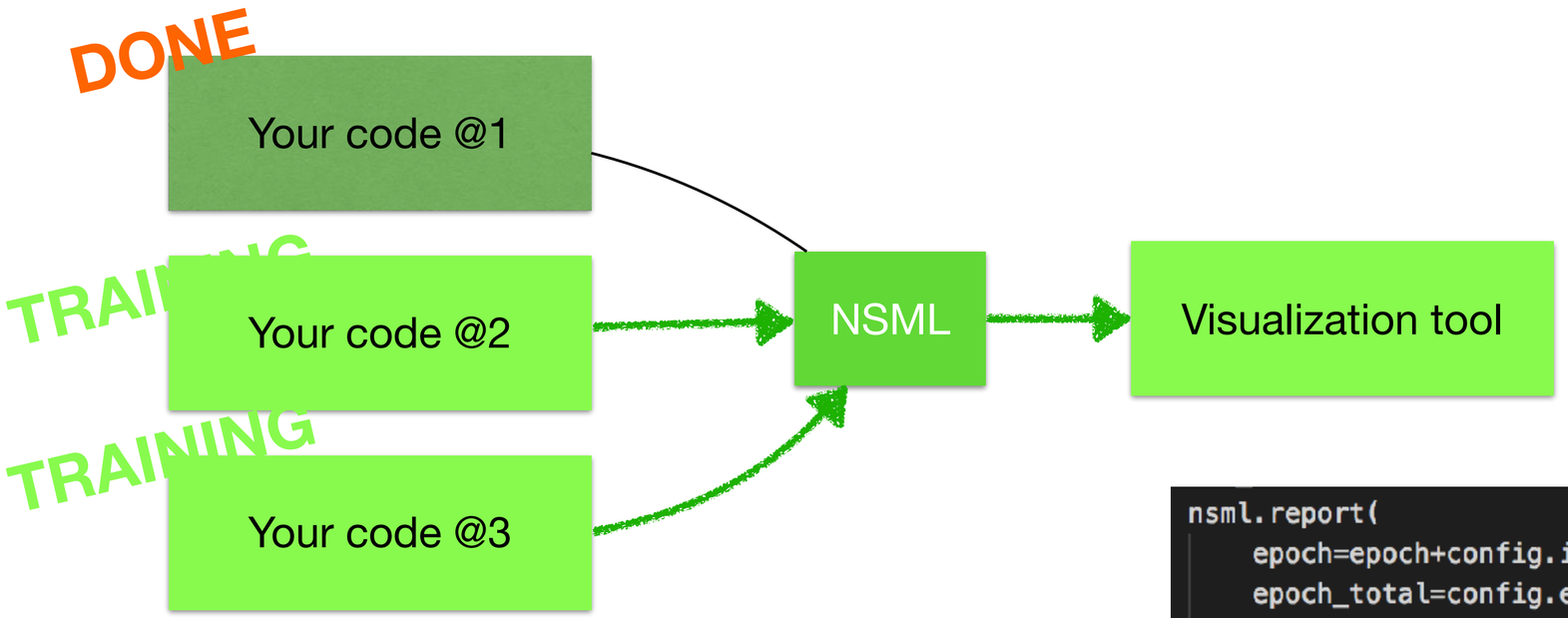
Need to Visualize

- Balance your brain to understand without effort

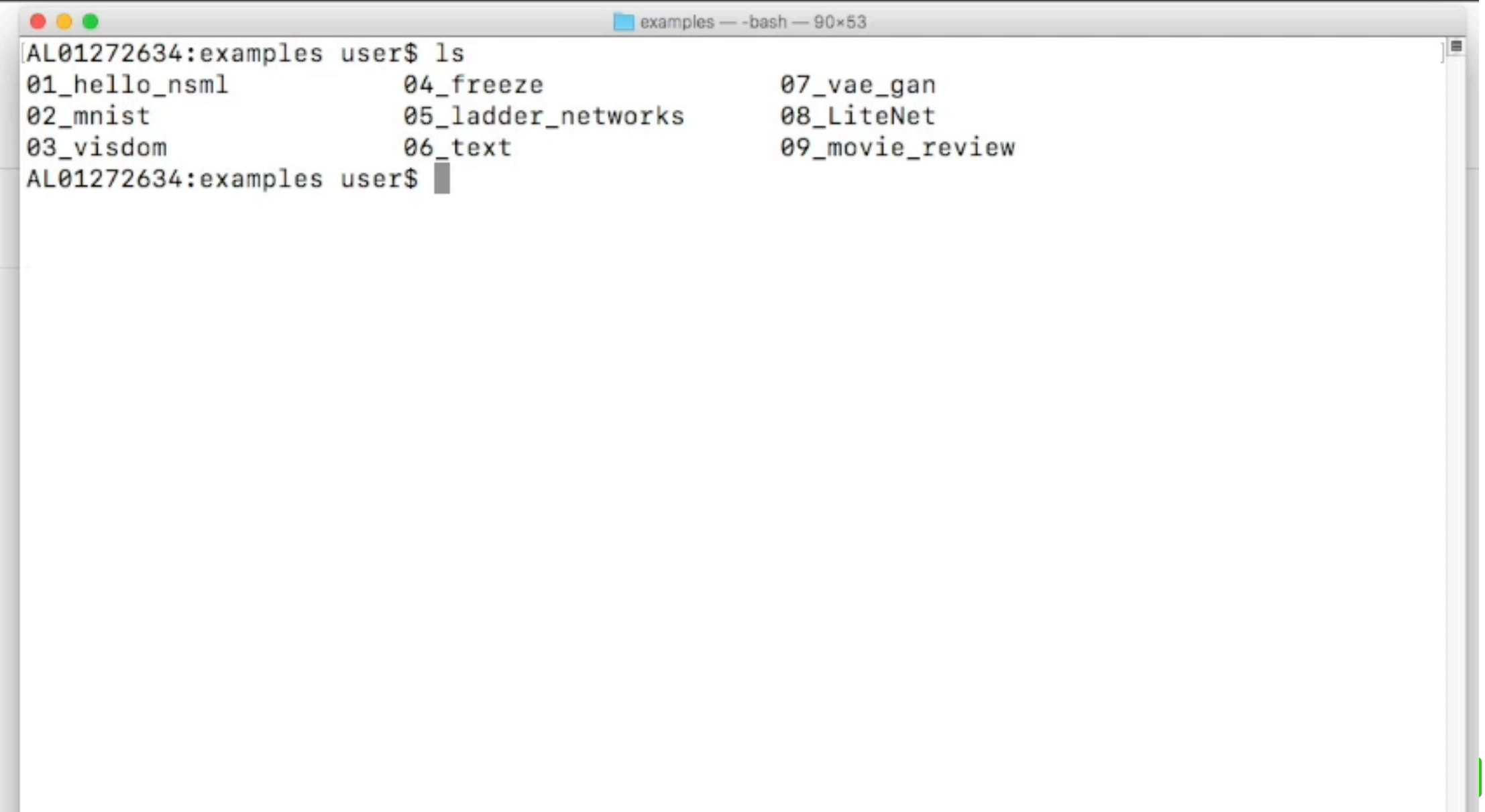


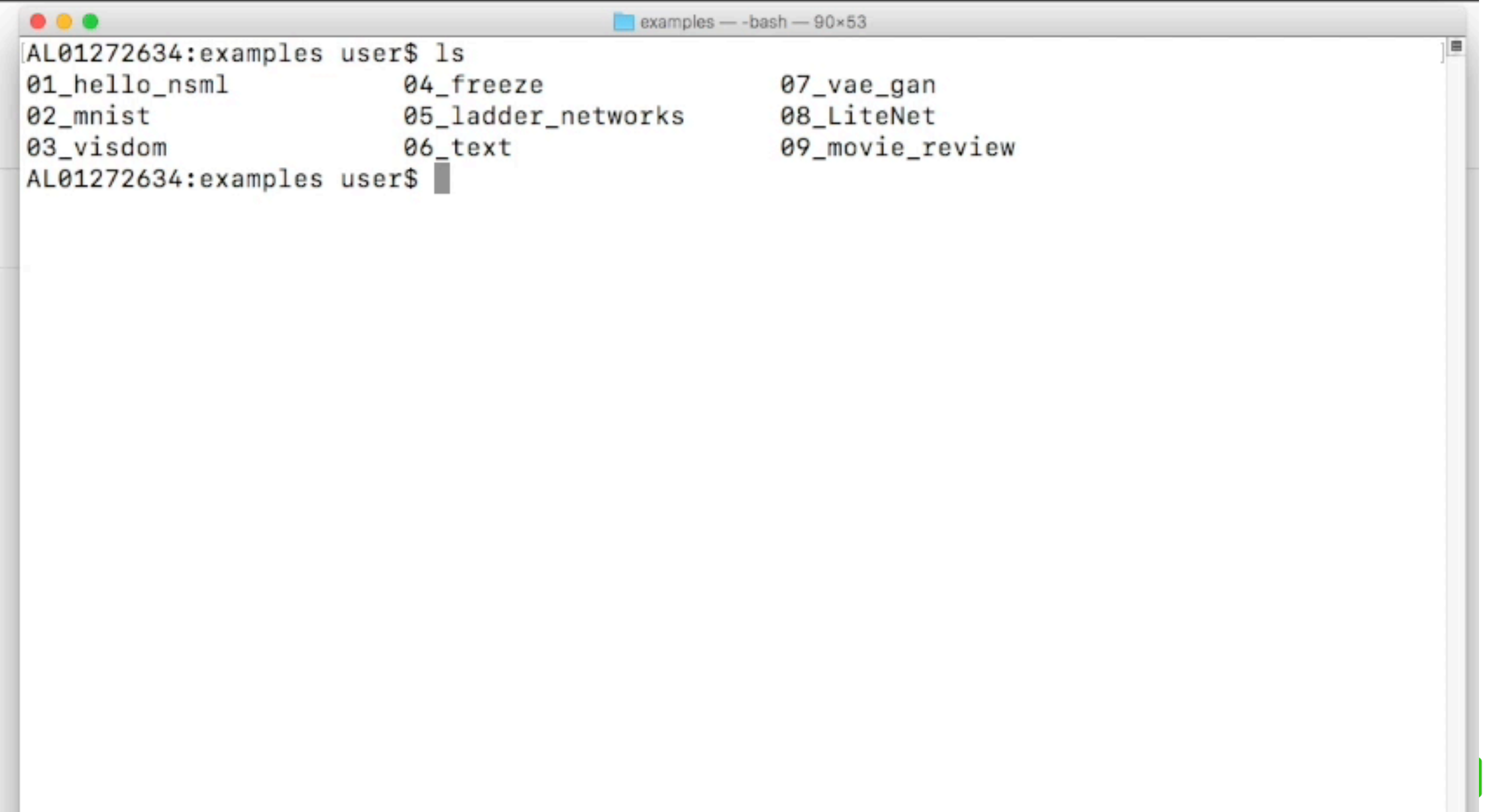
https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/data-visualization-for-human-perception

Flexible Analysis




```
nsml.report(
    epoch=epoch+config.iteration,
    epoch_total=config.epochs,
    iter=iter_idx,
    iter_total=total_length,
    batch_size=batch_size,
    train__loss=running_loss / num_runs,
    train__accuracy=running_acc / num_runs,
    scope=locals()
)
```


examples — -bash — 90×53
AL01272634:examples user$ ls
01_hello_nsml 04_freeze 07_vae_gan
02_mnist 05_ladder_networks 08_LiteNet
03_visdom 06_text 09_movie_review
AL01272634:examples user$


examples — -bash — 90×53
AL01272634:examples user$ ls
01_hello_nsml 04_freeze 07_vae_gan
02_mnist 05_ladder_networks 08_LiteNet
03_visdom 06_text 09_movie_review
AL01272634:examples user$

NAVER NSML

@nsml

Datasets › Sessions

Search ...

- bidaf_experiments
  a month ago · 390.67 MB
- CelebA_128
  24 days ago · 216.74 MB
  - rev : 12
    running · a minute ago
    gpu : 3
    args : main.py
  - rev : 10
    running · 4 minutes ago
    gpu : 3
    args : main.py
- cohn_kanade
  13 days ago · 1.04 GB
- mnist_torch
  a month ago · 127.8 MB
- movie_review
  14 hours ago · 606.08 MB
- searchqa
  a month ago · 710.01 MB
- SketchDB
  15 days ago · 42.83 MB
- snli
  a month ago · 946.33 MB

CelebA_128

Terminal | Graph

CelebA_128

Tensorboard

TensorBoard — SCALARS | IMAGES | AUDIO | GRAPHS | DISTRIBUTIONS | HISTOGRAMS | EMBEDDINGS | TEXT

Write a regex to create a tag group

Show data download links

Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis

STEP | RELATIVE | WALL

Runs

Write a regex to filter runs

- KR99114/CelebA_128/10
- KR99114/CelebA_128/12

TOGGLE ALL RUNS

/tmp/tensorboard

loss 5

loss/d_loss_fake

loss/d_loss_real

loss/g_loss_fake

loss/g_loss_kl

loss/g_loss_rec

Name	Smoothed	Value	Step	Time	Relative
KR99114/CelebA_128/10	0.1308	0.1764	130.0	Fri Oct 13, 13:18:41	9s

misc 1

misc/lr

NAVER NSML

@nsml

Datasets › Sessions

Search ...

- bidaf_experiments
  a month ago · 390.67 MB
- CelebA_128
  24 days ago · 216.74 MB
  - rev : 12
    running · a minute ago
    gpu : 3
    args : main.py
  - rev : 10
    running · 4 minutes ago
    gpu : 3
    args : main.py
- cohn_kanade
  13 days ago · 1.04 GB
- mnist_torch
  a month ago · 127.8 MB
- movie_review
  14 hours ago · 606.08 MB
- searchqa
  a month ago · 710.01 MB
- SketchDB
  15 days ago · 42.83 MB
- snli
  a month ago · 946.33 MB

CelebA_128

Terminal | Graph

CelebA_128

Tensorboard

TensorBoard — SCALARS | IMAGES | AUDIO | GRAPHS | DISTRIBUTIONS | HISTOGRAMS | EMBEDDINGS | TEXT

Write a regex to create a tag group

Show data download links

Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis

STEP | RELATIVE | WALL

Runs

Write a regex to filter runs

- KR99114/CelebA_128/10
- KR99114/CelebA_128/12

TOGGLE ALL RUNS

/tmp/tensorboard

loss 5

loss/d_loss_fake

loss/d_loss_real

loss/g_loss_fake

loss/g_loss_kl

loss/g_loss_rec

Name	Smoothed	Value	Step	Time	Relative
KR99114/CelebA_128/10	0.1308	0.1764	130.0	Fri Oct 13, 13:18:41	9s

misc 1

misc/lr

Dynamic Control Flow

Typical training loop

Forward pass
Backward pass
Communicate to NSML

```
nsml.report(
    epoch=epoch+config.iteration,
    epoch_total=config.epochs,
    iter=iter_idx,
    iter_total=total_length,
    batch_size=batch_size,
    train__loss=running_loss / num_runs,
    train__accuracy=running_acc / num_runs,
    scope=locals()
)
```

NSML

Command queue		
1	model	Watch a variable
2	change_lr(0.2)	Change a hyper parameter on the fly
3	nsml.save(‘quick’)	Save current snapshot
4	nsml.load(424)	Load saved snapshot
5	vis.image(model.generate(2))	Generate an image to visdom
…	….	…

NAVER

LINE

Terminal Graph

KR18284/mnist_torch/40 Terminal

```
Train Epoch: 4 [1920/60000 (3%)] Loss: 0.294512 Aux Loss: 0.071541
Train Epoch: 4 [2560/60000 (4%)] Loss: 0.440668 Aux Loss: 0.126457
Train Epoch: 4 [3200/60000 (5%)] Loss: 0.419388 Aux Loss: 0.078211
Train Epoch: 4 [3840/60000 (6%)] Loss: 0.378146 Aux Loss: 0.067423
Train Epoch: 4 [4480/60000 (7%)] Loss: 0.398869 Aux Loss: 0.082014
Train Epoch: 4 [5120/60000 (9%)] Loss: 0.718725 Aux Loss: 0.170943
Train Epoch: 4 [5760/60000 (10%)] Loss: 0.430032 Aux Loss: 0.096298
Train Epoch: 4 [6400/60000 (11%)] Loss: 0.363388 Aux Loss: 0.083657
Train Epoch: 4 [7040/60000 (12%)] Loss: 0.435897 Aux Loss: 0.102749
Train Epoch: 4 [7680/60000 (13%)] Loss: 0.313308 Aux Loss: 0.091727
Train Epoch: 4 [8320/60000 (14%)] Loss: 0.303983 Aux Loss: 0.064584
Train Epoch: 4 [8960/60000 (15%)] Loss: 0.378267 Aux Loss: 0.097526
```

```
>>>
```

NAVER LINE

Terminal Graph

KR18284/mnist_torch/40 Terminal

```
Train Epoch: 4 [1920/60000 (3%)] Loss: 0.294512 Aux Loss: 0.071541
Train Epoch: 4 [2560/60000 (4%)] Loss: 0.440668 Aux Loss: 0.126457
Train Epoch: 4 [3200/60000 (5%)] Loss: 0.419388 Aux Loss: 0.078211
Train Epoch: 4 [3840/60000 (6%)] Loss: 0.378146 Aux Loss: 0.067423
Train Epoch: 4 [4480/60000 (7%)] Loss: 0.398869 Aux Loss: 0.082014
Train Epoch: 4 [5120/60000 (9%)] Loss: 0.718725 Aux Loss: 0.170943
Train Epoch: 4 [5760/60000 (10%)] Loss: 0.430032 Aux Loss: 0.096298
Train Epoch: 4 [6400/60000 (11%)] Loss: 0.363388 Aux Loss: 0.083657
Train Epoch: 4 [7040/60000 (12%)] Loss: 0.435897 Aux Loss: 0.102749
Train Epoch: 4 [7680/60000 (13%)] Loss: 0.313308 Aux Loss: 0.091727
Train Epoch: 4 [8320/60000 (14%)] Loss: 0.303983 Aux Loss: 0.064584
Train Epoch: 4 [8960/60000 (15%)] Loss: 0.378267 Aux Loss: 0.097526
```

```
>>>
```

NAVER LINE

CLI

- Base of advanced features like save, load, infer, …

```
/tmp$ nsml run -d mnist_torch
Session KR18284/mnist_torch/12 is running
/tmp$ nsml exec KR18284/mnist_torch/12 model
Net (
  (conv1): Conv2d(1, 10, kernel_size=(5, 5), stride=(1, 1))
  (conv2): Conv2d(10, 20, kernel_size=(5, 5), stride=(1, 1))
  (conv2_drop): Dropout2d (p=0.5)
  (fc1): Linear (320 -> 50)
  (fc1_bn): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True)
  (fc2): Linear (50 -> 10)
  (conv2_bn): BatchNorm2d(20, eps=1e-05, momentum=0.1, affine=True)
)
```

Bring Your Own Workspace

- (Almost) Nothing to learn
- Cached (Fast)

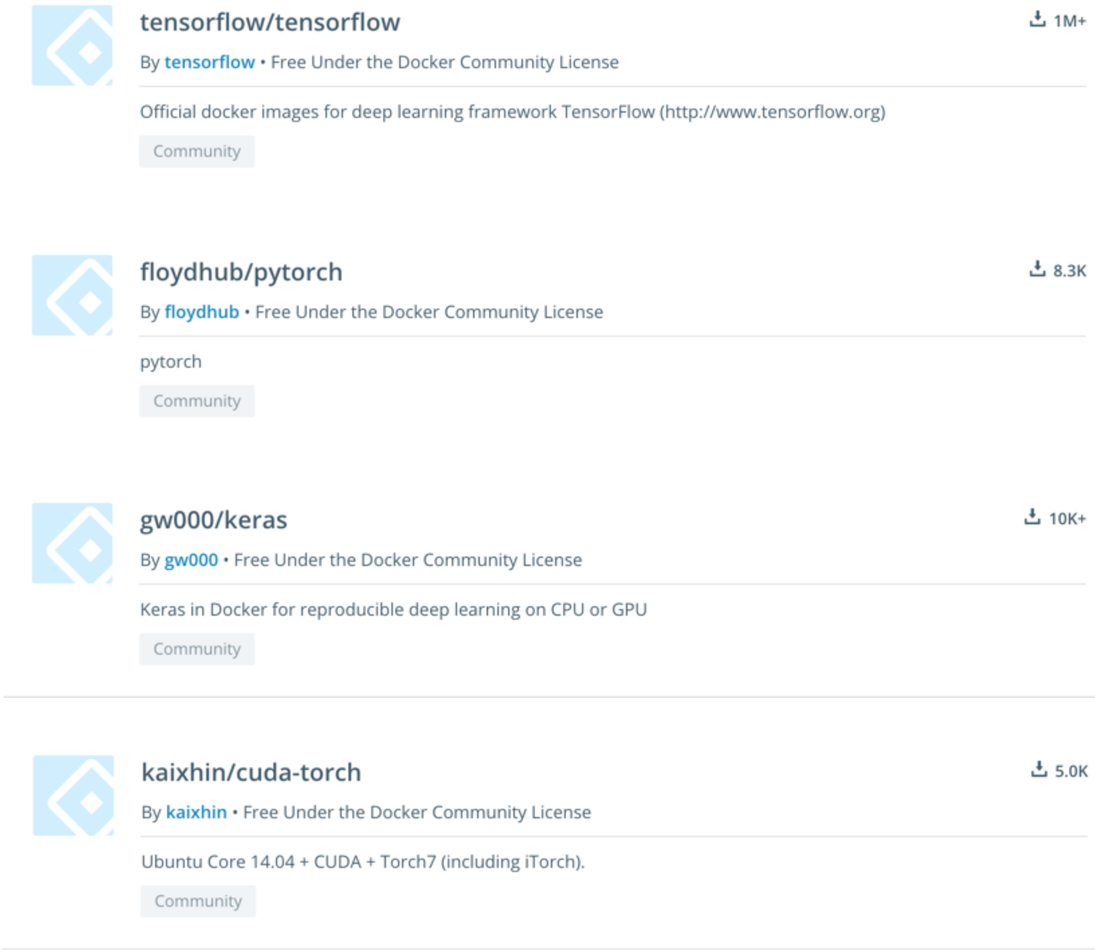
Bring Your Own Workspace

- (Almost) Nothing to learn
- Cached (Fast)

```
#nsml: nakosung/pytorch:latest-gpu-py3
from distutils.core import setup
setup(
    name='nsml example 07 VAE GAN',
    version='1.0',
    description='ns-ml',
    install_requires =[
        'visdom',
        'pillow'
    ]
)
```

NAVER LINE

No Framework Lock-in

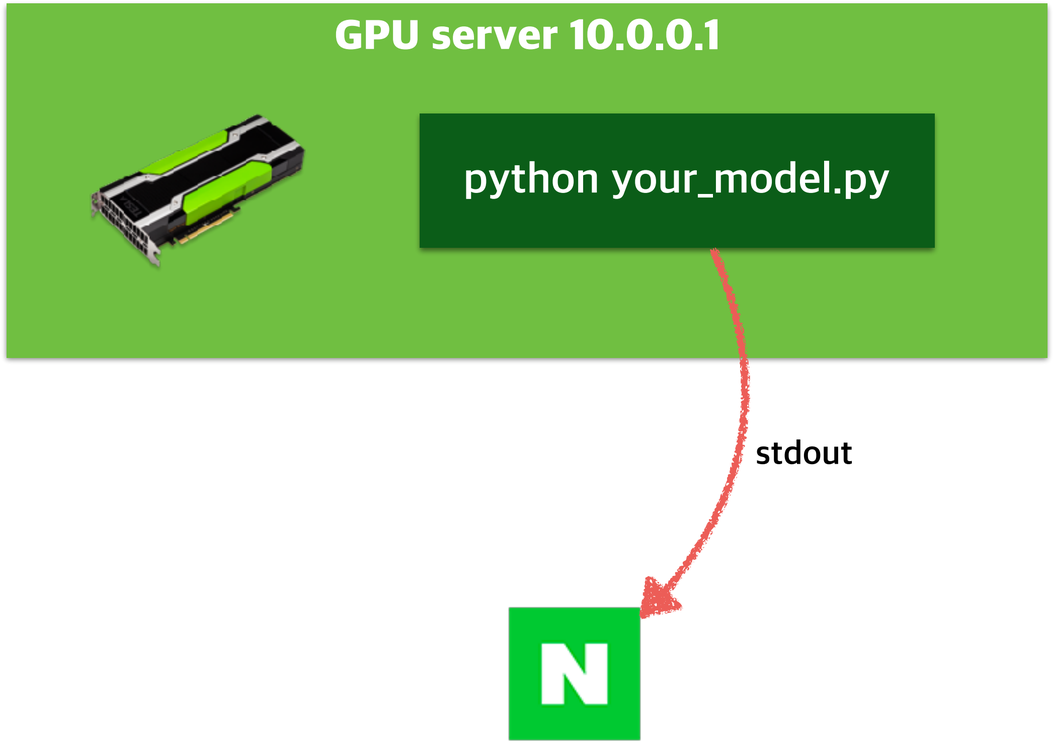


NAVER

LINE

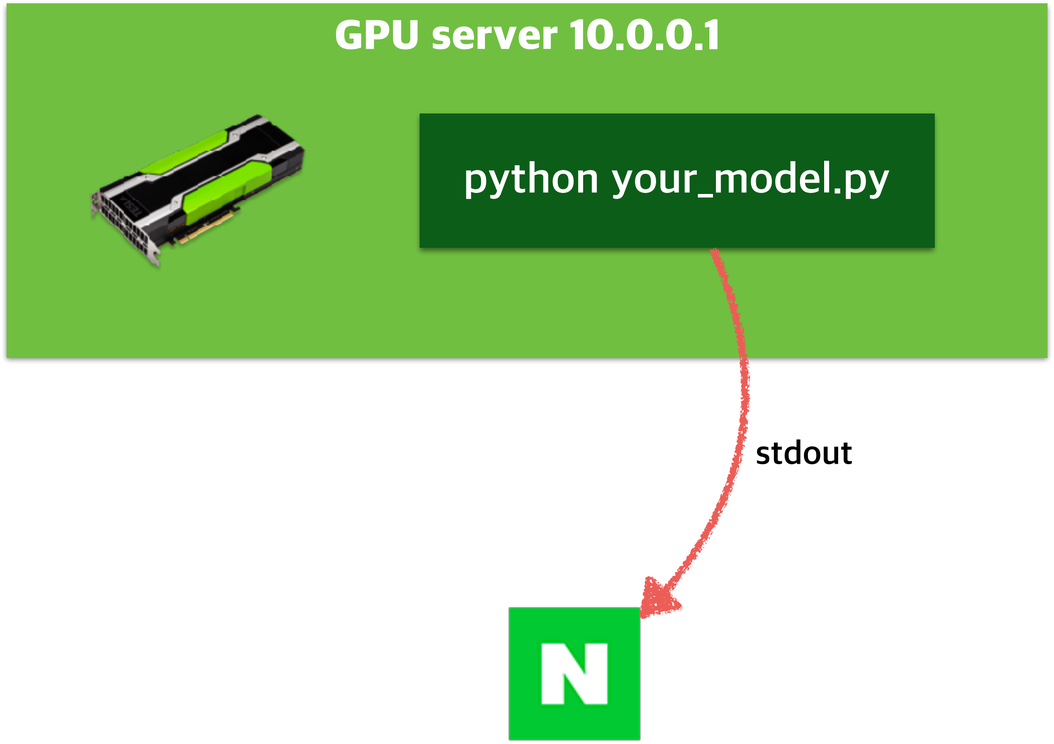
Interactive Mode

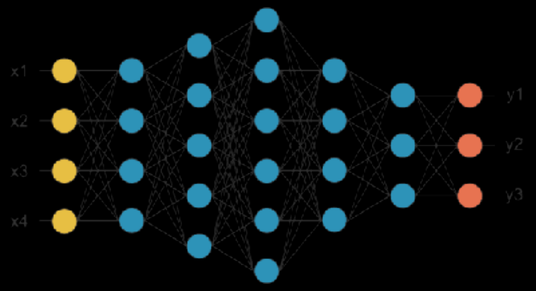
```
/app/examples/02_mnist$ n
```



Interactive Mode

```
/app/examples/02_mnist$ n
```

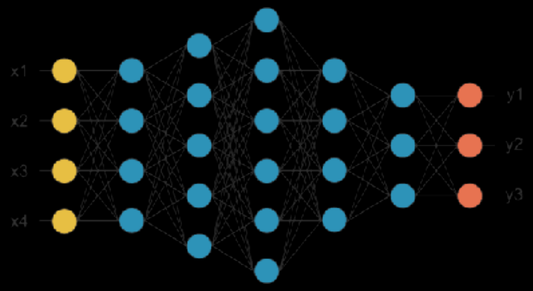




NAVER

LINE

Pragmatic Research



NAVER

LINE

NAVER NSML

@nsml

Datasets > Sessions > Models

cohn_kanade
13 days ago · 1.04 GB

mnist_torch
a month ago · 127.8 MB

rev : 14
exited(0) · 8 minutes ago
gpu : 1
args : main.py

SAVED MODEL · 9

epoch	9
epoch_total	10
reltime	250.97309064865112
test/accuracy	98.76
test/loss	5.8848886299133305
walltime	1507869330.1965399

SAVED MODEL · 8

epoch	8
epoch_total	10
reltime	225.81492924690247
test/accuracy	98.73
test/loss	5.98912923336029
walltime	1507869305.050199

SAVED MODEL · 7

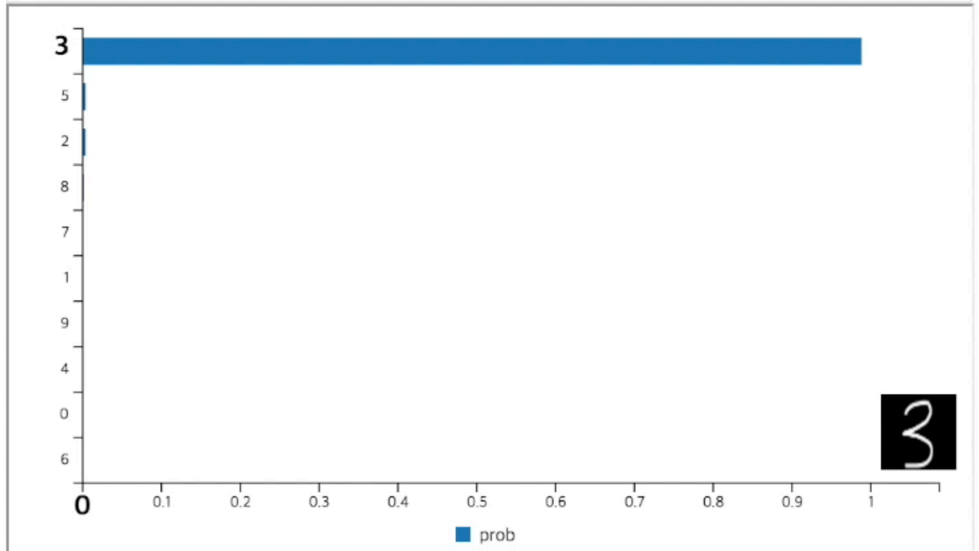
epoch	7

mnist_torch

Infer

Input

Output



Auto OFF

Result →

NAVER NSML

@nsml

Datasets > Sessions > Models

cohn_kanade
13 days ago · 1.04 GB

mnist_torch
a month ago · 127.8 MB

rev : 14
exited(0) · 8 minutes ago
gpu : 1
args : main.py

SAVED MODEL · 9

epoch	9
epoch_total	10
reltime	250.97309064865112
test/accuracy	98.76
test/loss	5.8848886299133305
walltime	1507869330.1965399

SAVED MODEL · 8

epoch	8
epoch_total	10
reltime	225.81492924690247
test/accuracy	98.73
test/loss	5.98912923336029
walltime	1507869305.050199

SAVED MODEL · 7

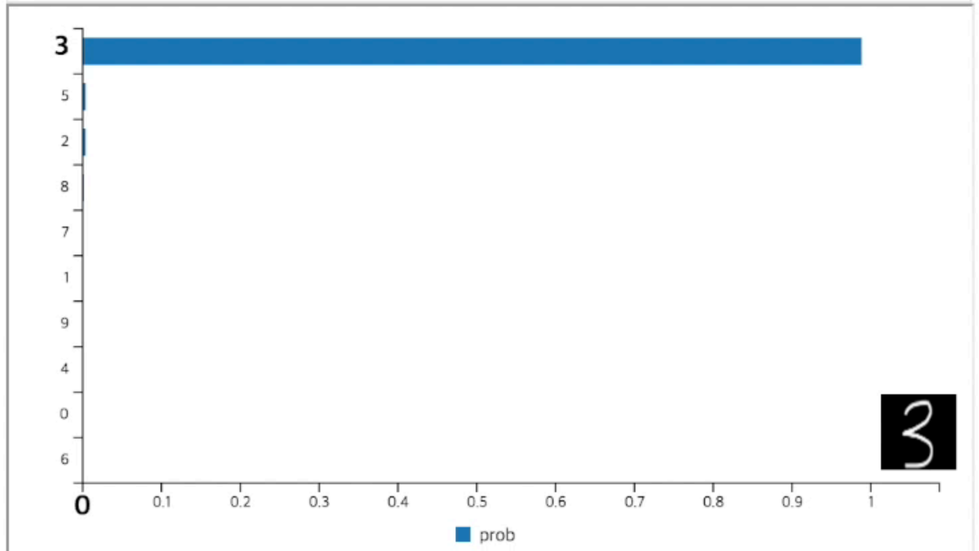
epoch	7

mnist_torch

Infer

Input

Output



Auto OFF

Result →

NAVER NSML

@nsml

Datasets > Sessions > Models

rev : 166

exited(0) · 2 days ago

gpu : 1

args : main_category1.py --litenet --layer_config 6 6 6 --use_dtcwt

SAVED MODEL · 160

epoch	160
reltime	8832.319140672684
test/err	1.046337817638266
test/loss	0.05320112881639079
test/top1	98.95366218236174
test/top5	99.85052316890882
total_epoch	160
walltime	1507716232.7313364

SAVED MODEL · 155

epoch	155
reltime	8545.214187383652
test/err	4.633781763826607
test/loss	0.2739498511914537
test/top1	95.3662182361734
test/top5	99.70104633781764
total_epoch	160
walltime	1507715945.6308603

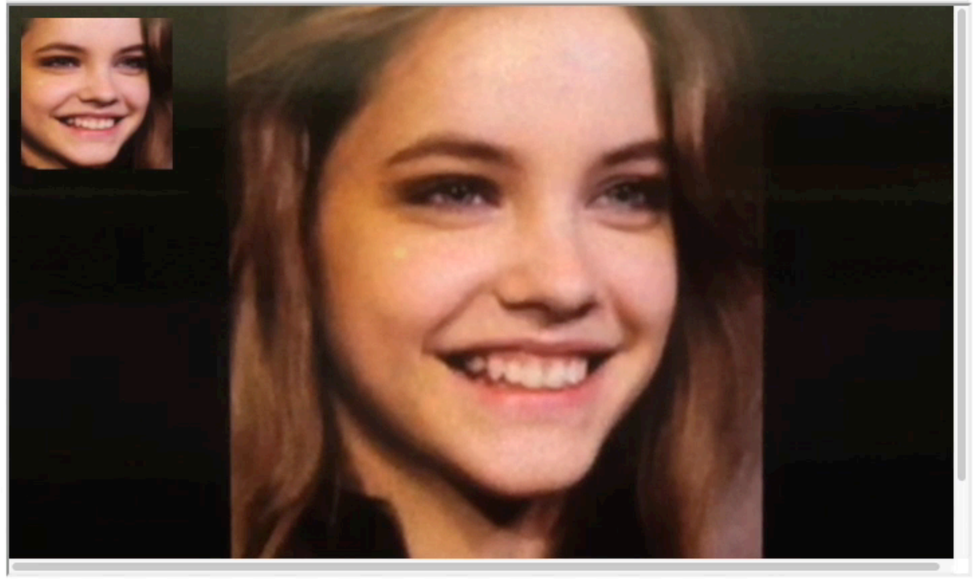
SAVED MODEL · 150

epoch	150
reltime	8259.60128736496

cohn_kanade

Infer

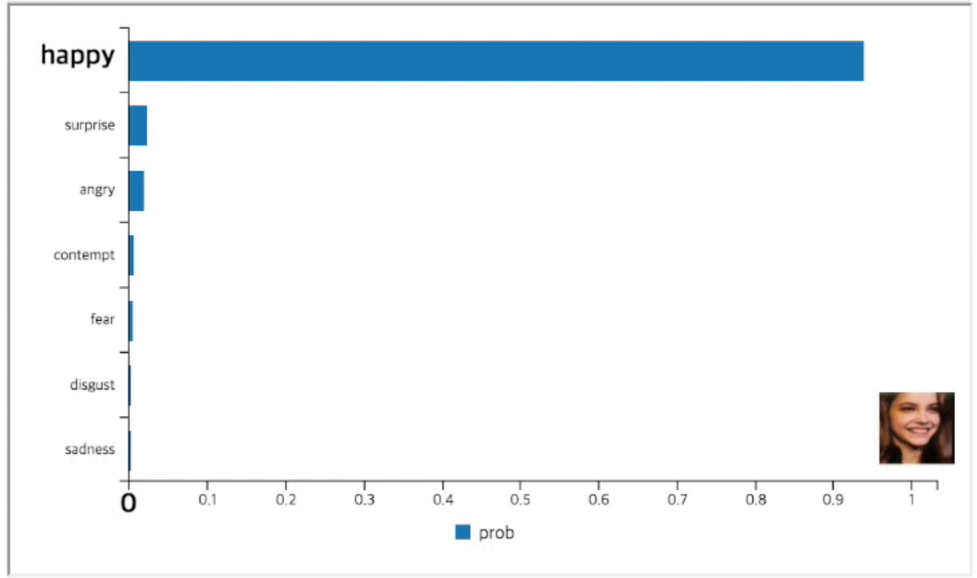
Input



Auto ON

Result →

Output



NAVER NSML

@nsml

Datasets > Sessions > Models

rev : 166

exited(0) · 2 days ago

gpu : 1

args : main_category1.py --litenet --layer_config 6 6 6 --use_dtcwt

SAVED MODEL · 160

epoch	160
reltime	8832.319140672684
test/err	1.046337817638266
test/loss	0.05320112881639079
test/top1	98.95366218236174
test/top5	99.85052316890882
total_epoch	160
walltime	1507716232.7313364

SAVED MODEL · 155

epoch	155
reltime	8545.214187383652
test/err	4.633781763826607
test/loss	0.2739498511914537
test/top1	95.3662182361734
test/top5	99.70104633781764
total_epoch	160
walltime	1507715945.6308603

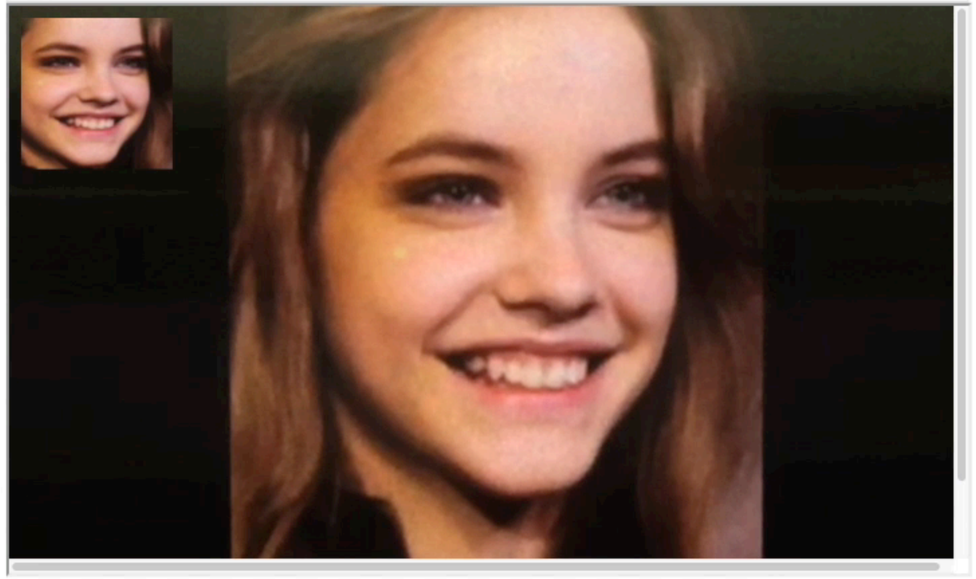
SAVED MODEL · 150

epoch	150
reltime	8259.60128736496

cohn_kanade

Infer

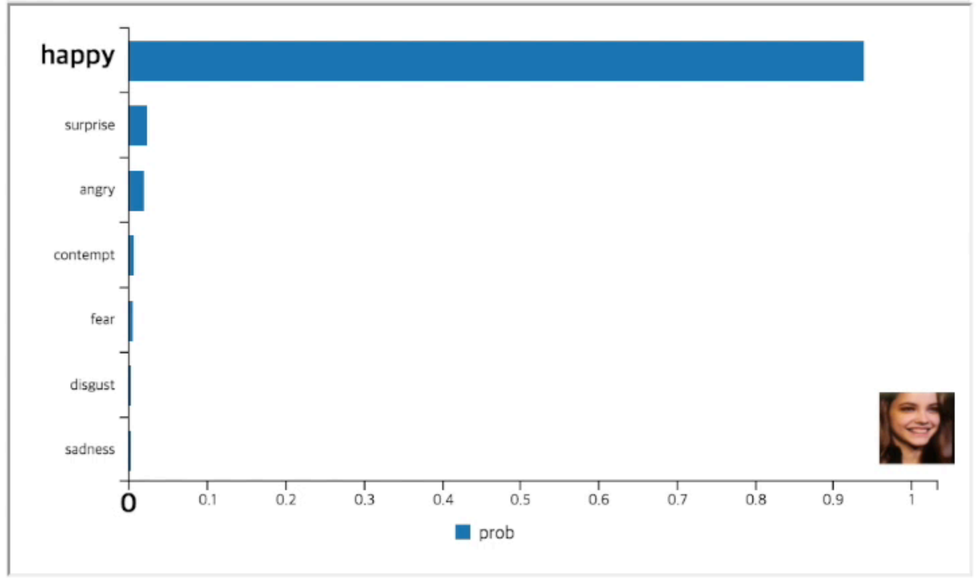
Input



Auto ON

Result →

Output



NAVER NSML

@nsml

Datasets > Sessions > Models

movie_review

19 hours ago · 606.08 MB

rev : 299

exited(0) · 20 hours ago

gpu : 1

args : lstm.py

desc : partial data only(10000) - preprocessing bug fixed

SAVED MODEL · 998

epoch	998
reltime	6224.894388914108
test/loss	6.172693252563477
train/loss	0.19896139577031136
walltime	1507819854.5514421

SAVED MODEL · 996

epoch	996
reltime	6212.374230384827
test/loss	6.237739562988281
train/loss	0.19785388931632042
walltime	1507819842.041744

SAVED MODEL · 994

epoch	994
reltime	6199.914450883865
test/loss	6.1442670822143555
train/loss	0.1978157013654709

movie_review

Infer

Input

기대안하고 봐야 할듯....

Auto ON

Result →

Output

★★½☆☆ 3

기대안하고 봐야 할듯....

NAVER NSML

@nsml

Datasets > Sessions > Models

movie_review

19 hours ago · 606.08 MB

rev : 299

exited(0) · 20 hours ago

gpu : 1

args : lstm.py

desc : partial data only(10000) - preprocessing bug fixed

SAVED MODEL · 998

epoch	998
reltime	6224.894388914108
test/loss	6.172693252563477
train/loss	0.19896139577031136
walltime	1507819854.5514421

SAVED MODEL · 996

epoch	996
reltime	6212.374230384827
test/loss	6.237739562988281
train/loss	0.19785388931632042
walltime	1507819842.041744

SAVED MODEL · 994

epoch	994
reltime	6199.914450883865
test/loss	6.1442670822143555
train/loss	0.1978157013654709

movie_review

Infer

Input

기대안하고 봐야 할듯....

Auto ON

Result →

Output

★★½☆☆ 3

기대안하고 봐야 할듯....

Collaboration and Competition
Leaderboard, CI-ML

NAVER LINE

New Workflow for ML Research

Collaboration and Competition
Leaderboard, CI-ML

NAVER LINE

Collaborative Research

- Easy to reproduce and extend other’s research.

```
/tmp$
```

NAVER

LINE

Collaborative Research

- Easy to reproduce and extend other’s research.

```
/tmp$
```

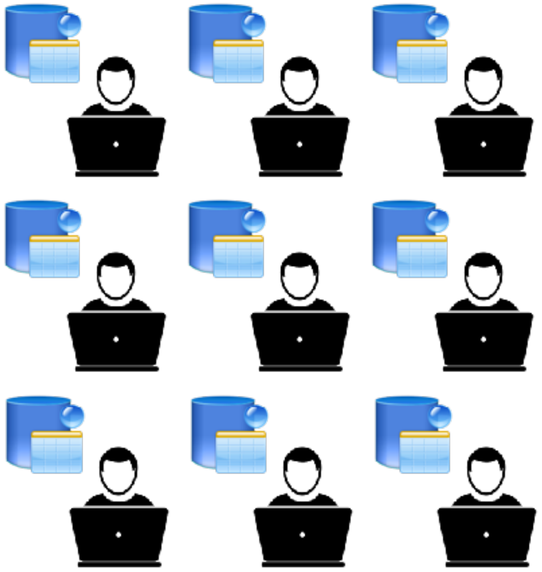
NAVER

LINE

Cohesive and Competitive

Dataset-centric environment

Models are ranked automatically



Standardized and Quantified

Easy to compete

Towards AutoML

NAVER

LINE

Cohesive and Competitive

Dataset-centric environment

Models are ranked automatically

Standardized and Quantified

Easy to compete

Towards AutoML

NAVER

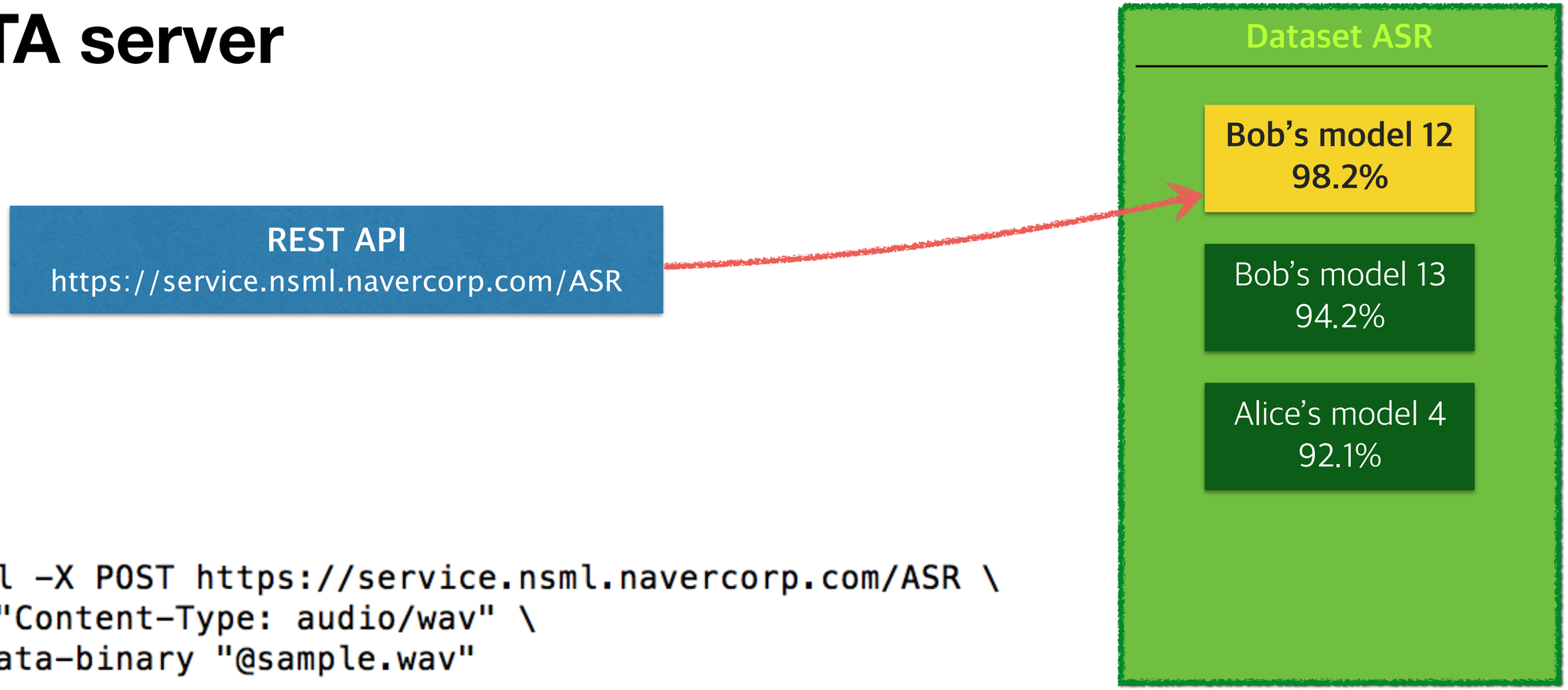
LINE

AutoML

- Quantitive model analysis makes ML workflow as a gym of AutoML

Seamless Connection to Services

SOTA server




```
$ curl -X POST https://service.nsml.navercorp.com/ASR \
  -H "Content-Type: audio/wav" \
  --data-binary "@sample.wav"
```

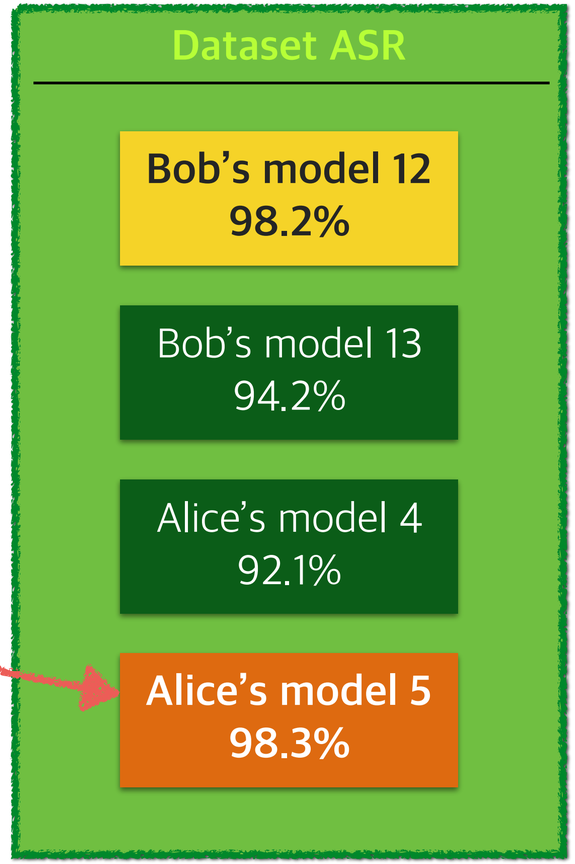
NAVER

LINE

Seamless Connection to Services

SOTA server

REST API
https://service.nsml.navercorp.com/ASR




```
$ curl -X POST https://service.nsml.navercorp.com/ASR \
  -H "Content-Type: audio/wav" \
  --data-binary "@sample.wav"
```

NAVER LINE

Q1. 2018

NAVER

LINE

https://research.clova.ai/nsml-alpha

Thank you

Several Hundreds of GPUs for this alpha (free)