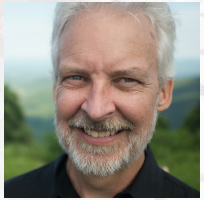
NLP SUMMIT

Presented by John Snow LABS

Ray for NLP



Dean Wampler

Principal Software Engineer

Domino Data Lab (formerly Anyscale)

@deanwampler

Domino Data Lab

DOMINO Products Solutions Customers Resources Partners Company See Demo Try Now

System-of-Record for Enterprise Data Science Teams

dominodatalab.com

Accelerate Research

Get self-serve access to the latest tools and scalable compute. Reuse past work and iterate more efficiently.

Learn More »

Centralize Infrastructure

Manage the availability of powerful data science resources in a secure and governed system-of-record.

Learn More »

Deploy and Monitor Models

Expedite model consumption with apps, APIs, and more – and ensure their accuracy for key decisions.

Learn More »

Unify Data Science Teams

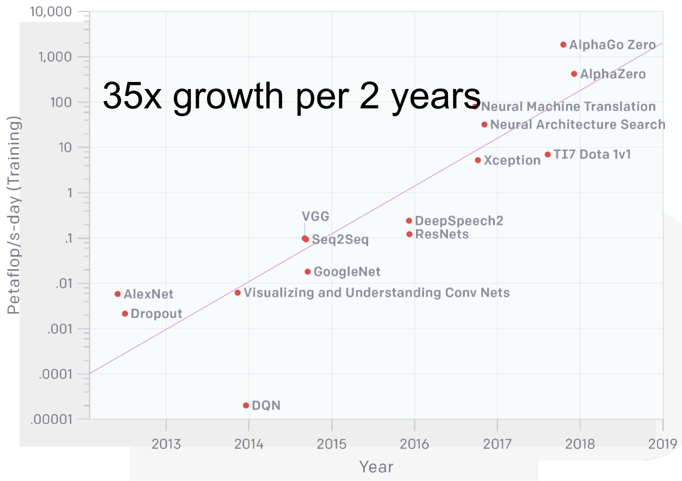
Make data science teams more productive and collaborative, and manage their work more efficiently.

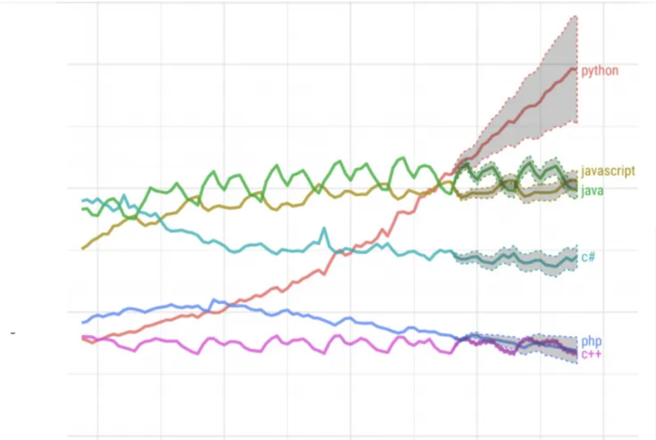
Learn More »

NLP SUMMIT

Why Ray

- Model sizes and compute requirements are growing rapidly.
- Python is the dominant data science programming language.
- ray.io





NLP SUMMIT

@deanwampler

How to scale Python in N easy steps!

```
def make_array(…):
    a = … # Construct a NumPy array
    return a


def add_arrays(a, b):
    return np.add(a, b)
…
```

The Python you know…

NLP SUMMIT

How to scale Python in N easy steps!

```python
@ray.remote
def make_array(…):
    a = … # Construct a NumPy array
    return a

@ray.remote
def add_arrays(a, b):
    return np.add(a, b)
…
```

Turn a function into a task.

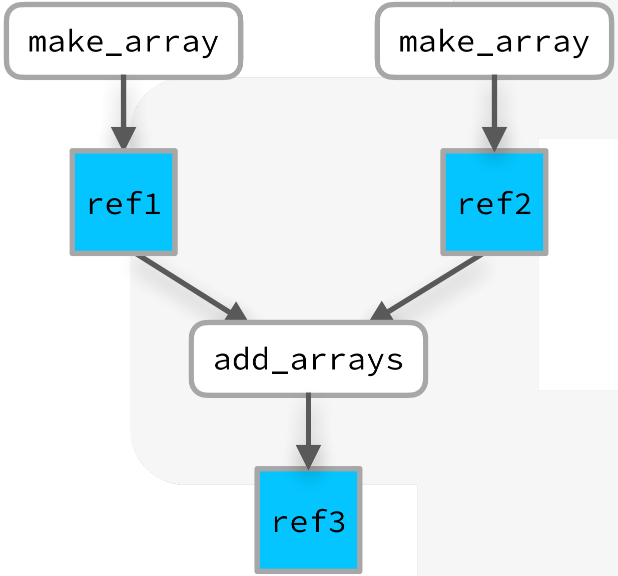
```python
# for completeness, start with:

import ray
import numpy as np

ray.init(…)
```

NLP SUMMIT

How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
```

Start a task with remote.

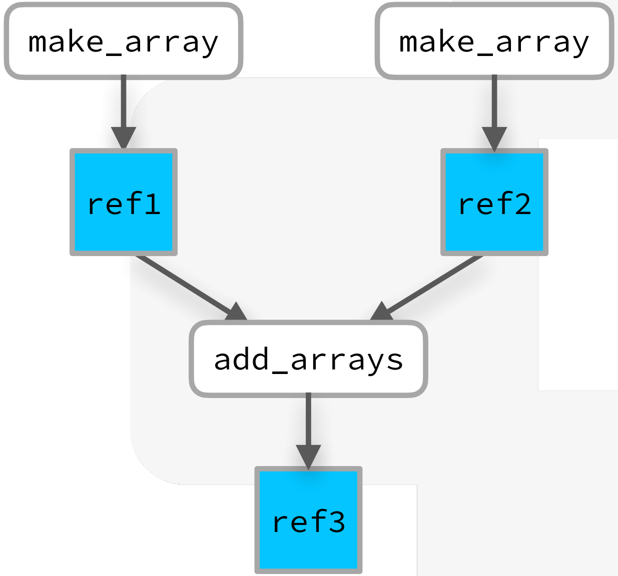


NLP SUMMIT

How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

Fetch the computed value



How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

No need to call ray.get() for these first!

Ray handles cluster scheduling, async computing

NLP SUMMIT

How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

What about Distributed state??

NLP SUMMIT

How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

```
class Counter(object):
    def __init__(self):
        self.value = 0
    def increment(self):
        self.value += 1
        return self.value
```

The Python classes you love

How to scale Python in N easy steps!

```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

```
@ray.remote
class Counter(object):
    def __init__(self):
        self.value = 0
    def increment(self):
        self.value += 1
        return self.value
    def get_count(self):
        return self.value
```

From class to “actor”

Must add a getter method

NLP SUMMIT

How to scale Python in N easy steps!

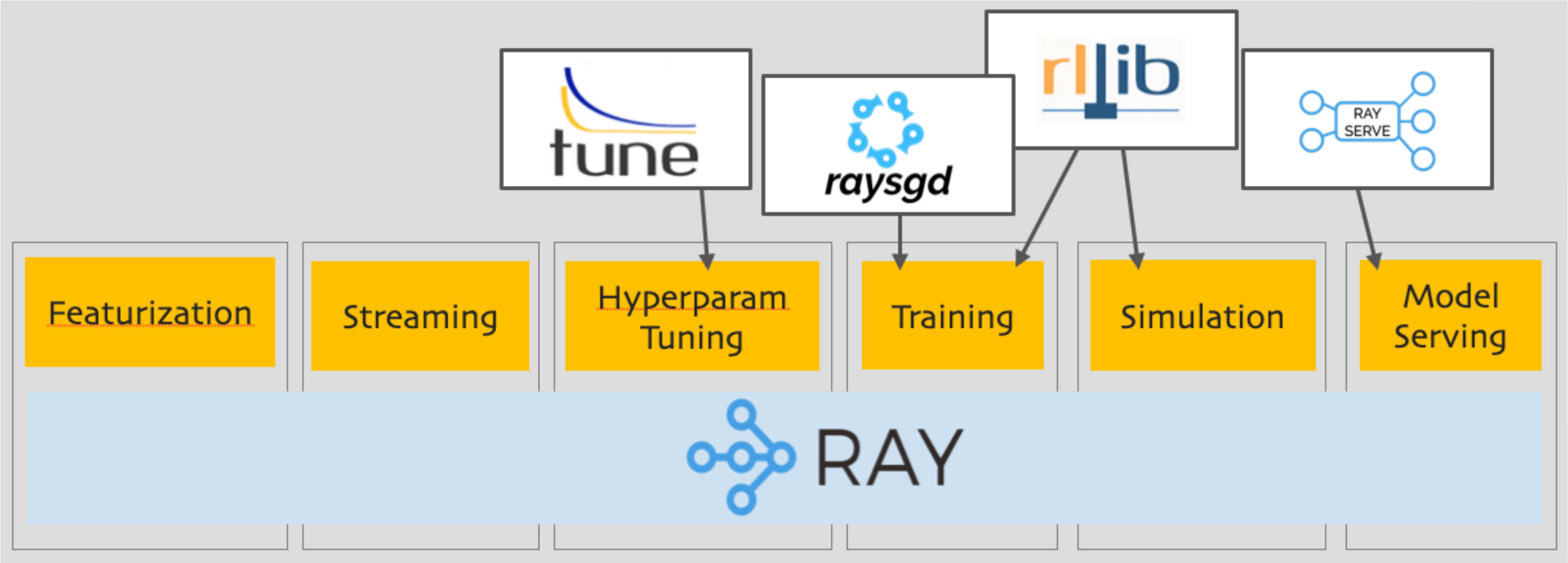
```
…
ref1 = make_array.remote(…)
ref2 = make_array.remote(…)
ref3 = add_arrays.remote(ref1, ref2)
array = ray.get(ref3)
```

```
…
c = Counter.remote()
ref4 = c.increment.remote()
rer5 = c.increment.remote()
ray.get([ref4,ref5]) # [1, 2]
```

Same idioms

NLP SUMMIT

But you may never use the Ray API…



[ray.io](ray.io)

NLP SUMMIT

@deanwampler

But you may never use the Ray API…

huggingface.co

Hugging Face Transformers

Since NLP model training is $$$$$, it's easier to use **transfer learning:**

- Start with a pre-trained model
- Add a few more layers
- Train for a few epochs for a particular application
- Profit?

Transformers

github.com/huggingface/transformers

Hugging Face Transformers

Well, hyper-parameter tuning is also expensive and it can be tricky.

- Avoiding local minima: arxiv.org/abs/1811.01088
- High variance in results common: github.com/pytorch/fairseq/blob/master/examples/roberta/wsc

🤗 Transformers

github.com/huggingface/transformers

credit: Thomas Wolf, *Transfer Learning in NLP: Concepts, Tools & Trends* (Ray Summit 2020)

Hugging Face Transformers

Using Ray Tune You can get 1.5% better results using *Bayesian Optimization*, 5% better using *Population-Based Training* for the same compute resources.

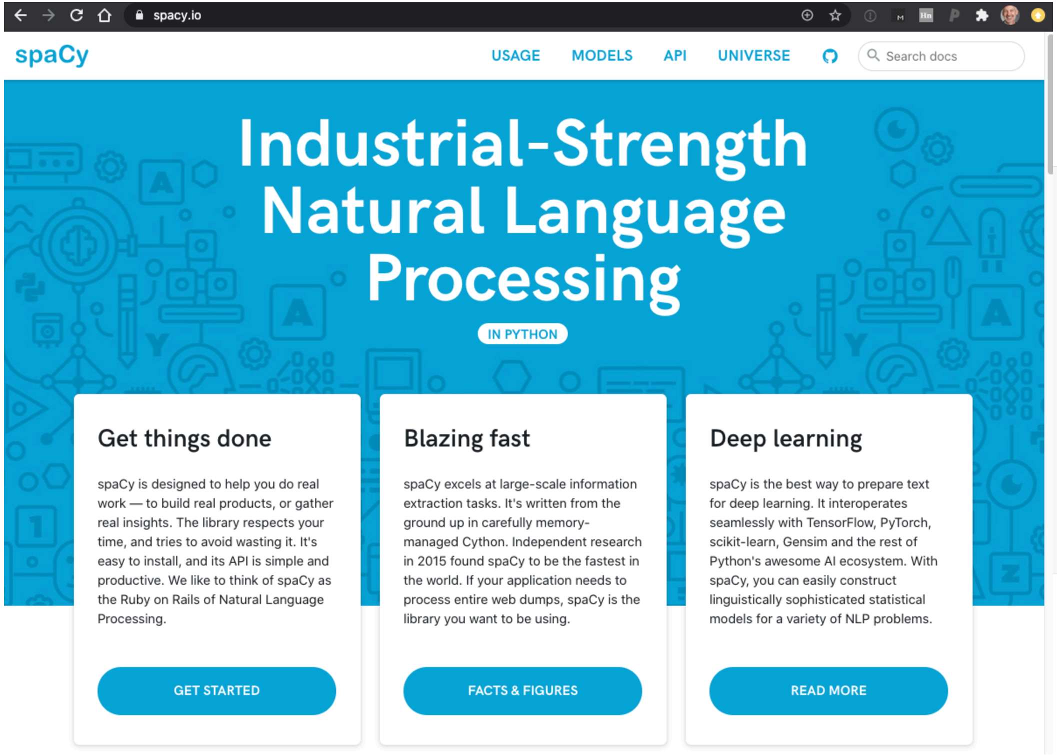
- See this blog post (Ray blog on Medium)

tune.io

Hyper-parameters:

- learning rate
- weight decay
- # of epochs
- per-GPU batch size

But you may never use the Ray API...



spacy.io

spaCy v3

spaCy v3 release will introduce a new integration with Ray, which will bring effortless parallel and distributed training to spaCy.

- github.com/explosion/spacy-ray
- Matthew Honnibal, Why spaCy Is Going with Ray (Ray Summit 2020)

explosion.ai

spaCy v3

```
$ python -m spacy train …
```

CLI without spacy-ray

```
$ pip install spacy-ray
$ python -m spacy ray train --n-workers 2 …

# “spacy ray pretrain” and “spacy ray parse”
# are planned.
```

Parallel and distributed training with spacy-ray

NLP SUMMIT

spaCy v3

spaCy v3 includes changes to the data model and some pipeline improvements. The Ray support is not a lot of code:

- Shard parameters into distributed state with Ray actors.
- Train on local shard.
- Asynchronously receive updates from other actors.
- Merge updates.
- Repeat...

CLI without spacy-ray

Parallel and distributed training with spacy-ray

NLP SUMMIT

Your uses of Ray for NLP?

Ray's flexible *task* model can be used for course- and fine-grained computation. The *actor* model makes "sharded", distributed state intuitive to manage. So, use it for:

- Tokenization and other data prep
- Distributed training: Ray Tune and Ray SGD for easier distributed TensorFlow and PyTorch
- Simple, scalable model serving with Ray Serve

ray.io

NLP SUMMIT

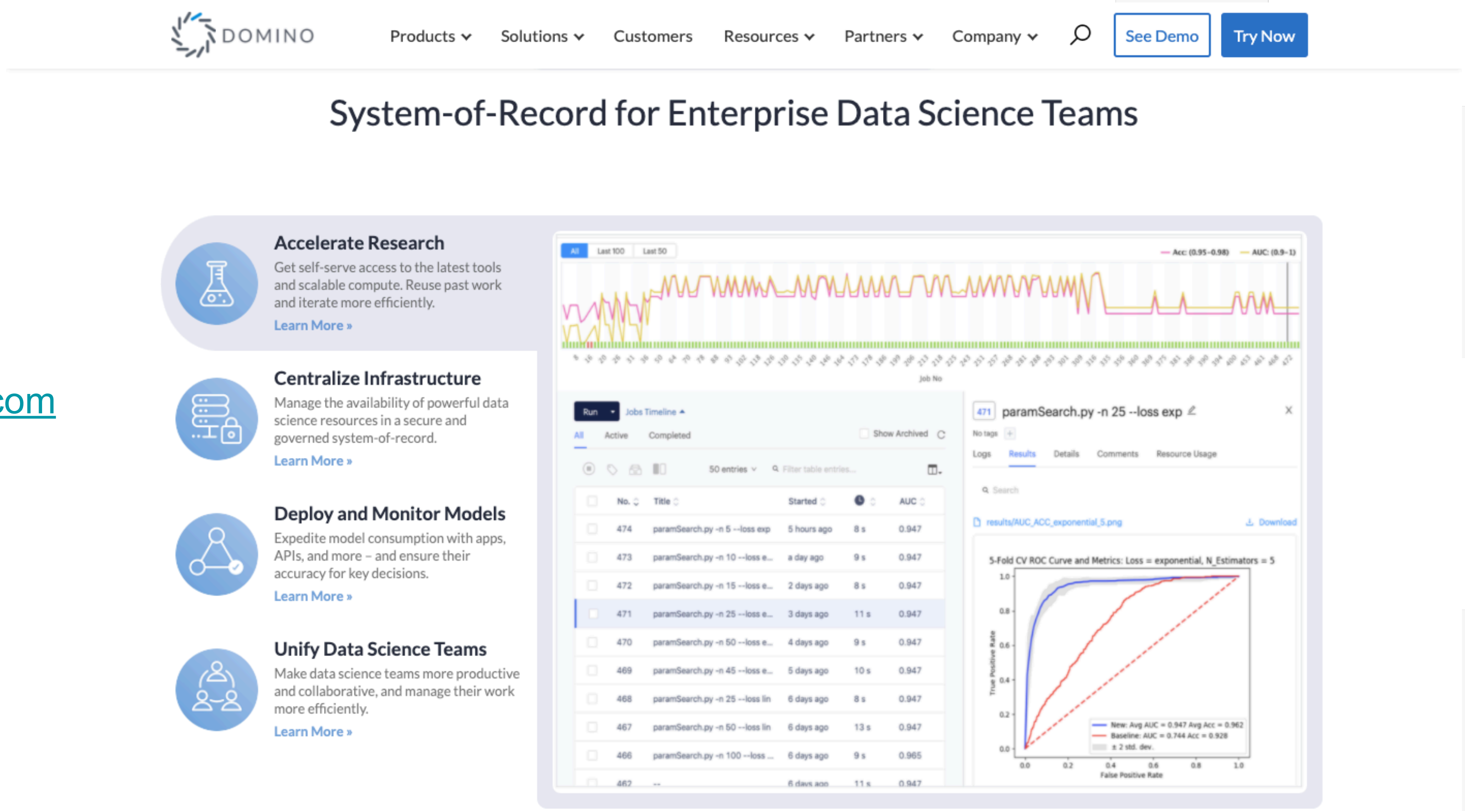
Ray in NLP

The creators of Hugging Face and spaCy and the how they use Ray. See the Anyscale blog and YouTube channel.

raysummit.org

Domino Data Lab

dominodatalab.com



NLP SUMMIT

Thanks for listening!

- ray.io
- raysummit.org
- dean@dominodatalab.com
- @deanwampler

NLP SUMMIT

Presented by John Snow LABS

Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.