

Automatic differentiation in ML

Where we are and where we should be going

Bart van Merriënboer¹², Olivier Breuleux¹, Arnaud Bergeron¹, Pascal Lamblin¹²

Mila¹, Google Brain²



Key contributions

- Where are we?
 - A critical survey of implementation approaches to machine learning frameworks from a compiler, programming language, system, and automatic differentiation perspective
- Where should we be going?
 - Myia is a prototype of a machine learning framework which brings the best of multiple worlds

Two paradigms of ML frameworks

Dataflow programming

```
x = tf.placeholder(tf.float32)
y = x * x
dx, = tf.gradients(y, x)
```

```
with tf.Session() as sess:
    dx_ = sess.run(
        dx, feed_dict={x: 3})
```

Operator overloading

```
def f(x):
    return x * x
```

```
df = grad(f)
dx = df(3)
```

Trade-offs

	PyTorch, TF Eager, Gluon	TensorFlow, Caffe2, MXNet
<i>AD implementation</i>	Operator overloading	Graph transformation
<i>AD overhead</i>	- Runtime	+ Ahead of time
<i>Runtime</i>	- Python interpreter	+ Custom runtime
<i>User interface</i>	+ Python	- API
<i>Program representation</i>	+ Linear trace (numeric only)	- Dataflow graph
<i>Generality</i>	+ Python	- Limited

Best of both worlds

	PyTorch, TF Eager, Gluon	TensorFlow, Caffe2, MXNet	Myia , Relay
<i>AD implementation</i>	Operator overloading	Graph transformation	Program transformation ¹
<i>AD overhead</i>	- Runtime	+ Ahead of time	+ Ahead of time
<i>Runtime</i>	- Python interpreter	+ Custom runtime	+ Custom runtime
<i>User interface</i>	+ Python	- API	± Python subset
<i>Program representation</i>	+ Linear trace (numeric only)	- Dataflow graph	+ Functional IR
<i>Generality</i>	+ Python	- Limited	± Python subset

¹To understand how our proposal differs from frameworks such as Tangent or TensorFlow for Swift, come see our poster!

Thank you!

Come talk to us at poster #94

See the code at github.com/mila-udem/myia