What is deep learning?
And

Comparing various deep learning learning frameworks

Dhruv Apte

(the-ethan-hunt)

About me

- The GIF guy
- Data scientist: ML, NLP, DL are my three musketeers
- https://github.com/the-ethan-hunt - That's my GitHub profile
- @apte_dhruv - Twitter profile
- Great with Python and MATLAB
- Okayish with R and Julia
- Near-zero with Java

What is deep learning?

- When you use an algorithm to train your computer to solve a pre-defined example and try it on a new example, you have got machine learning
- Deep learning is that part of machine learning where you use a model of computing like a human brain!
- Get your kid to wear clothes!(To differentiate between AI, ML and DL)

The big thing for today!

BRACE YOURSELVES
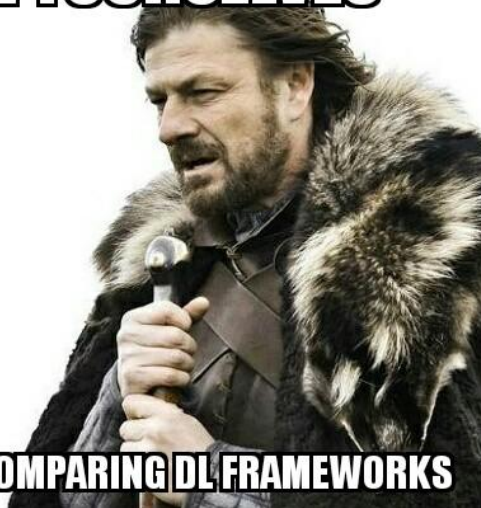
WE ARE NOW COMPARING DL FRAMEWORKS

1) Caffe

❖ Caffe was developed by Yangqing Jia at UC Berkeley back when he was doing his Ph.D

❖ Dr. Jia is currently at FAIR developing Caffe2

❖ It's open sourced on GitHub

❖ Advantages:
  ➢ Really nice interface for Python or MATLAB
  ➢ Awesome for image classification(it has built-in models so you don't need to write a single line of code)
  ➢ It's really fast!

❖ Cons:
  ➢ You cannot define your own layers
  ➢ Don't try to tinker with the code
  ➢ You cannot go outside its comfort zone

2) Chainer

❖ Chainer was developed by scientists from University of Tokyo
❖ Advantages:
➢ No CUDA or GPU jargon to scare you: Chainer works even on 32-bit windows system
➢ Really helpful community; they are on Slack!
➢ Was initially built for dynamic computation and NLP

❖ Disadvantages:
➢ Slow computing power as compared to recent frameworks
➢ Low quality of visualization
➢ Not so used by deep learning community

3) Keras

❖ Developed by Francois Chollet
❖ Advantages:
  ➢ Great for starters: Keras is usually the framework used in introduction courses
  ➢ Researchers love it!
  ➢ Easy deployment
  ➢ Multiple backend engines- Tensorflow, CNTK, Thean
❖ Disadvantages:
  ➢ Lack of pre-trained models
  ➢ Difficult debugging
  ➢ Tinkering with the code is difficult

arXiv mentions, October 2017
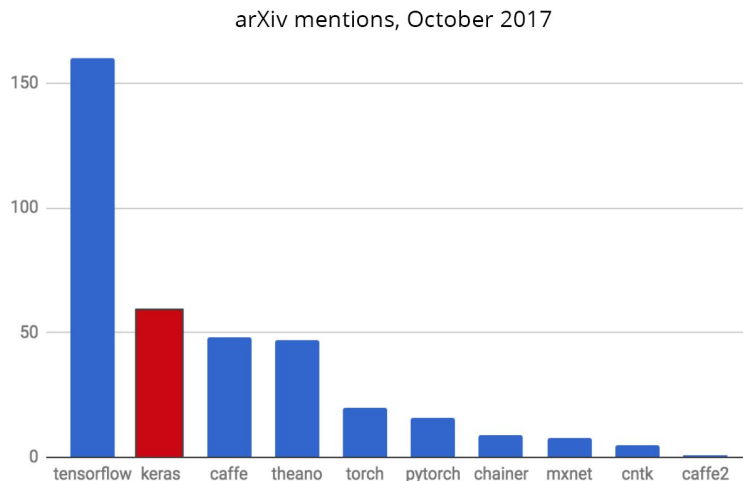
4) CNTK

❖ Developed by Microsoft Research
❖ Advantages:
  ➢ Twice as fast as TensorFlow
  ➢ CNTK can be easily extended from Python for layers and learners
  ➢ Very powerful C++ and Python APIs
❖ Disadvantages:
  ➢ Not much used apart from Microsoft
  ➢ Lesser outreach than Torch or TensorFlow

5) Torch/ PyTorch

❖ Developed by FAIR
❖ Advantages:
  ➢ The **BEST** user interface
  ➢ Torch has been preferred over and over again by industry due to its comfortable use
  ➢ You don't get confused over the theoretical aspects or the architecture
  ➢ You can define, change and execute nodes as you go, no special session interfaces or placeholders. Overall, the framework is more tightly integrated with Python language
  ➢ Very easy to debug

❖ Disadvantages:
  ➢ Not so-great visualization(Tensorboard rules!)
  ➢ Lacks good deployment and distributed training

6) TensorFlow

----> [Deep Wars](#)

Episode I: The Death of Theano

- Google released its framework known as TensorFlow after several cases of problems with Chainer, CNTK and Keras.
- Later Theano stopped accepting contributions and active maintenance and TensorFlow stole the limelight
- It could even run on Android or iOS.
- With the release of TensorBoard, TF left all other frameworks back in terms of data visualization
- TF became popular due to its community and even has an MOOC!
- TF suddenly became the biggest Jedi!

Episode II: The Empire Strikes Back

- However, it was soon discovered that TF had a lot of problems
- It had really very large binaries(20 MB+) and you dare not tinker with them!
- The documentation was not as good as expected(C++)
- Too low level to use comfortably for rapid prototyping, yet too high level to use comfortably in cutting edge research or in production environments that are resource constrained.
- Not great for debugging at all!
- Comparison!
- Less explicit functions

Thanks to :

1. Google Developers Group Baroda
2. @Brorlandi for the Star Wars creator
3. Star Wars for being so awesome!

Questions are welcome!

May the neural network be with you!