

10 Machine Learning Frameworks for Your Consideration in 2020

Originally published at <https://www.iflexion.com/blog/machine-learning-frameworks>

Published: May 6, 2020 | By Martin Anderson



Evaluating a machine learning library for commercial deployment can mean choosing between the latest trends and the most reliable and proven architectures. By the time an ML library has gained market traction and widespread developer availability, it's often struggling to assimilate the most recent innovations in the machine learning sector.

Conversely, more recent frameworks may have only a nascent user and developer base, with roadmaps that depend upon an industry take-up and economy of scale that may never arrive.

The domain and scope of our project should be the primary deciding factor along with commercial licensing costs. In some cases, where goals are very predefined, a mature legacy architecture may be a better solution than the current 'buzz' technologies.

Therefore, here is the comparison of ten notable machine learning frameworks covering the most popular fields in [AI consulting](#), featuring a diversity of maturity levels and rates of industry adoption.

These are 10 #MachineLearning frameworks rocking it for 2020 projects

Accord.NET



Language: C#

Sectors: Computer vision, audio analysis

License: Gnu Lesser Public License, version 2.1

Accord.NET is a .NET machine learning library for image-based workflows such as [facial recognition](#), object tracking, and audio analysis. Its dedicated audio module features a large variety of methods, interfaces, and arguments.

Support for clustering, regression and classification makes Accord.NET one of the most accessible machine learning frameworks available, and its capabilities can be explored through numerous sample applications.

Accord.NET offers ample, well-organized documentation that makes implementing a new machine learning project straightforward.

Considerations

Accord.NET comes with all the framework overhead of the .NET approach, and thus may compare unfavorably to TensorFlow in terms of performance. Although its exclusive support for .NET frameworks brings a Unity API, this limits the potential for Accord.NET to utilize popular libraries without the burden of interpretive API layers.

Furthermore, this integration with a popular meta-framework is offset by its low industry take-up. Despite an impressive body of documentation, troubleshooting resources can also prove scarce in comparison to other libraries listed here.

Apache Spark MLlib



Sectors: Classification and filtering

Language: Scala

License: Apache License 2.0

MLlib is a dedicated machine learning Library for the low-latency Spark framework. MLlib facilitates supervised and unsupervised training, and benefits from Spark's RAM-driven performance boost. Machine learning algorithms supported are: regression, collaborative filtering, clustering, and binary classification.

MLlib features a speedy generalized linear algebra engine called Breeze. However, Breeze does not support certain popular features and functions in machine learning without the aid of third-party engines.

MLlib is interoperable with NumPy via Python, and can make use of diverse file systems and clusters including [Hadoop](#), Kubernetes, and Apache Mesos.

Considerations

MLlib's speed benefits and friendly high-level APIs are offset by limited scope and frequent dependence on slower systems such as Mahout over Hadoop. Unless you're committing to a Hadoop deployment (which brings related performance penalties), MLlib can take quite a bit of setting up.

Read more about Apache Spark in [our Spark vs Hadoop review](#).

Caffe



Sectors: Image segmentation, image classification

Language: C++

License: BSD 2-Clause license

Caffe is a low-latency C++ library for image classification and segmentation, with bindings for MATLAB and [Python development](#) and an expressive architecture supporting the development of convolutional neural networks (CNNs).

An imperative and low-level resource, Caffe can iterate through 60 million images per day, utilizing GPU or CPU resources as necessary.

A useful range of pre-trained models is available in Caffe's model zoo repository. Sectors covered include robotics, speech and visual classification frameworks.

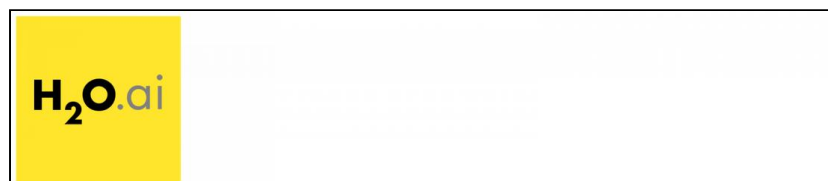
Considerations

Caffe is difficult to set up for Recurrent Neural Networks, and may need significant configuration to work with more recent architectures, since it is not a generalized project and lacks friendly high-level APIs.

However, this same factor cuts down on framework overhead, making Caffe one of the most performant and persistently popular solutions in image-based research.

Read more about the framework in [our Caffe vs TensorFlow matchup](#).

H2O.ai



Sectors: Statistical analysis, business intelligence

Language: Java (with bindings for Python and R)

License: Apache License 2.0

H2O.ai is among the current leaders in the impetus to automate scalable machine learning by abstracting the development process into a GUI-driven 'dashboard' paradigm. H2O.ai features 'driverless AI', wherein a pipeline is developed algorithmically from a range of available ML approaches.

Models are automatically validated, tuned, selected and deployed, with a wide range available, from deep learning through to generalized linear approaches.

The web-based interface H2O Flow offers command line input but also a programming-free interactive environment that creates explorable graphical objects from a guided query process.

Scripts written in R or CoffeeScript are supported for model creation and analysis, as are Jupyter notebooks and cross-user collaboration.

Considerations

Data imported to H2O.ai must be labelled, and pre-processing may be necessary to create suitably well-structured data. It may be easier to identify and exploit new insights from data reservoirs through less restrictive architectures, such as Hadoop. The limits of standalone H2O.ai are implied in its explicit support for Spark through Sparkling Water.

Keras



Sectors: Image segmentation, image classification, NLP

Language: Python

License: MIT

Keras is a user-friendly high-level API for the development of neural networks. In itself it supports TensorFlow and CNTK, MXNet and Theano.

However, Keras has received reciprocal support in CNTK since 2016, and has been the official API for TensorFlow since 2017. Keras is frequently run as a facilitating user-space above those two platforms, as well as R and non-NVIDIA GPU-based machine learning deployments via PlaidML. Keras is also directly supported on Apple's CoreML on iOS and the Android mobile platform via the TensorFlow Android runtime.

Keras can also deploy neural networks in its own right, though without the advantages of GPU acceleration.

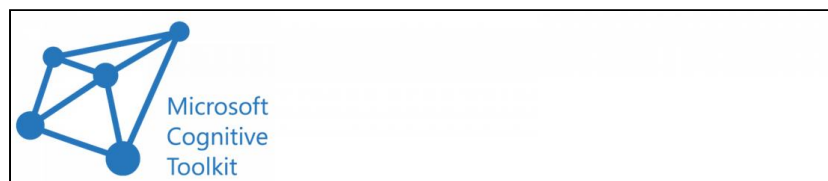
Keras is extensible and modular, and comes with a set of predefined layer types, as well as seven datasets covering NLP and various sub-sectors of image recognition, and ten models for image classification.

Considerations

The benefit of Keras' friendly abstractions are offset by the necessity for error-handling outside of the target framework, where feedback on errors is more consistent.

Certain ML algorithms can also run slower on the GPU than the CPU in Keras in certain circumstances.

Microsoft Cognitive Toolkit (CNTK)



Sectors: Image, speech and handwriting recognition

Language: C++/Python

License: MIT

Microsoft Cognitive Toolkit (CNTK) is a deep learning framework that facilitates a range of neural network types, including convolutional neural networks, recurrent neural networks and feed-forward DNNs via directed graphs.

CNTK has full support for CUDA 10, and features automated differentiation and stochastic gradient descent with parallel implementation. CNTK performs well, scales easily and doesn't require intervention into low-level code. Besides C++, CNTK also supports Python and is accessible via a CLI.

CNTK's support for Keras ensures ongoing interoperability with the most popular competing frameworks, and it also comes its own model editor called BrainScript.

One of the first machine learning frameworks to support the Open Neural Network Exchange (ONNX) model interchange format, CNTK has support for MATLAB, Caffe 2, Keras, Chainer, PyTorch, and several other formats.

Considerations

Cessation of further development after V2.7 was announced in 2019, as Microsoft moves to more abstracted self-automating systems via ML.NET. However, CNTK is arguably 'feature complete'.

If mobile or ARM support is required in your project, you will need to look elsewhere.

PyTorch



Sectors: Computer vision, NLP, speech recognition

Language: Python, C++, CUDA

License: 3-clause BSD

PyTorch is a Facebook-led open initiative built over the original Torch project and now incorporating Caffe 2. Offering wide applicability and high industry take-up, PyTorch has a distinct foothold in NLP, [computer vision software](#) and facial recognition research, thanks to Facebook's vast quantities of user-generated data.

Since PyTorch is imperative, end-users can run queries and code immediately without developing a full build. PyTorch also features scalable distributed training and TorchScript, which can export models to non-Python production environments.

PyTorch enables deep neural networks and tensor computing workflows similar to TensorFlow and leverages the GPU likewise.

The native Optim module allows automatic optimization of deployed neural networks, with support for most of the popular methods. PyTorch can track and replay all operations through its AutoGrad automatic differentiation package, enabling on-the-fly gradient generation.

Considerations

PyTorch is a less mature, more research-oriented library than some of its rivals. Its documentation and help resources can prove limited or poorly organized.

SpaCy



Sectors: NLP

Language: Python/Cython

License: MIT

SpaCy is a Python library particularly well-suited to [text analysis](#). It's optimized for high-volume applications and operates at celebrated speed. It's written in Cython, a Python superset that can run in low-level and high-performing C-based frameworks.

Surprisingly, SpaCy has no in-built functionality for sentiment analysis. Instead, this functionality must be laid over SpaCy's provision for syntactic parsing and chunking, Stanford NER and word vectors.

SpaCy features a range of templated NLP models including classification, named entity recognition, and part-of-speech (POS) tagging. However, these are relatively inflexible, and may not suit all needs.

Considerations

In terms of speed, rival frameworks have caught up somewhat with SpaCy over the last five years, albeit by sacrificing useful functionality, such as extended granularity when handling contractions.

Though it comes with higher hardware requirements than some competitors, and though it lacks architectural flexibility, SpaCy remains one of the most robust and high-performing Python libraries in the sector, as well as one of the most accessible and consistent.

Read more about SpaCy in our overview of [Python sentiment analysis frameworks](#).

Stanford CoreNLP



Sectors: NLP

Language: Java (1.8+ for later versions)

License: GNU General Public License v2/3

Stanford Core NLP is a regularly updated collection of Java libraries that offers a pre-installed set of versatile linguistic analysis tools, including a dedicated sentiment analysis component.

Stanford CoreNLP can be used in conjunction with a variety of other languages from C# to ZeroMQ, and offers several compatibility options for various iterations of Python. The package can be run as a web service or else through various APIs. Its features include parsing, co-reference resolution, bootstrapped pattern learning, named entity recognition, and part-of-speech tagging.

Support is provided via Stanford's mailing list, through a dedicated support email address, and through the Stack Overflow community.

Considerations

A Java programming environment can be difficult to deal with directly, as may be necessary for certain operations, while CoreNLP's latency remains a contested topic.

Additionally, CoreNLP's mix of v2/3 licensing may require commercial fees payable for distributions that include certain parts of CoreNLP code.

Read more about Stanford Core NLP in our overview of [Python sentiment analysis frameworks](#).

TensorFlow



Sectors: Computer vision, NLP, speech recognition, Convolutional Neural Networks (CNNs)

Language: C++/Python/CUDA

License: Apache License 2.0

Thanks to its broad applicability, cross-platform support, close ties to GPU acceleration on the NVIDIA platform, and high-level industry backing, TensorFlow has become one of the most important libraries in machine learning over the last ten years.

Developed by Google, TensorFlow features support for GPU, CPU and Google's own ASIC Tensor processing unit (TPU). In addition to its use in [computer vision applications](#), TensorFlow supports NLP, Time Series Forecasting, speech recognition, and text classification.

It's a user-friendly, powerful solution that both benefits from and contributes to NVIDIA's ascendancy in the machine learning space.

Considerations

TensorFlow owes its versatility to a complex architecture that can be challenging to navigate, and which is completely committed to NVIDIA through the cuDNN libraries, despite fringe efforts to provide more diverse GPU support.

However, extensive high-level industry support tends to balance out TensorFlow's occasional performance shortfalls against lighter frameworks.

Read more about the framework in [our Caffe vs TensorFlow comparison](#).

AT A GLANCE

	Core Language	Sector	First release
Accord.NET	C#	Computer vision, audio analysis	2017
Spark	Scala	Classification and filtering	2014
Caffe	C++	Image segmentation, image classification	2016
H2O.ai	Java	Statistical analysis, business intelligence	2017
Keras	Python	Image segmentation, image classification, NLP	2015
CNTK	C++	Image, speech, handwriting recognition	2016
PyTorch	Python	Computer vision, NLP, speech recognition	2016
SpaCy	Python	NLP	2015
Stanford CoreNLP	Java	NLP	2010
TensorFlow	C++	Computer vision, NLP, speech recognition	2015
IFLE  ION			