Keras resources

Keras-Official docs

Keras is a Python library for deep learning that runs on TensorFlow, Theano or CNTK. http://keras.io/

Keras is the ideal library for rapid experimentation. Currently, the most popular deep learning library for Theano and TensorFlow is Keras, which provides a high-level interface to both Theano and TensorFlow.

Keras was developed and maintained by François Chollet, a Google engineer using four guiding principles:

- **Modularity**: A model can be understood as a sequence or a graph alone. All the concerns of a deep learning model are discrete components that can be combined in arbitrary ways.
- **Minimalism**: The library provides just enough to achieve an outcome. No frills, maximum readability. Ease of use.
- **Extensibility**: New components are easy to add, and old ones are well-use within the framework, intended for researchers to try out wild new ideas.
- **Python**: No separate model files with custom file formats. Everything is in native Python.

- Keras Documentation
- Keras on Github
- Official Keras examples

Keras examples — General & Basics

- Develop Your First Neural Network in Python With Keras Step-By-Step (Jason Brownlee on May 24, 2016)

In this post, you discovered how to create your first neural network model using the powerful Keras Python library for deep learning.

Specifically, you learned how to take your first steps from a simple linear to a neural network or deep learning model, step-by-step including:

1. How to load data.
2. How to define neural networks in Keras.
3. How to compile a Keras model using the efficient numerical backend.
4. How to fit a model on data.
5. How to evaluate a model on data.

- Handling Keras with TensorFlow backend
- 9 Ways to Get Help with Deep Learning in Keras (Jason Brownlee on August 1, 2016 in Deep Learning)
- Keras Tutorial: The Ultimate Beginner's Guide to Deep Learning in Python

This tutorial is pretty good. It provides some good links to some basic concepts of machine learning.

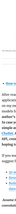
- Introduction to Python Deep Learning with Keras (Jason Brownlee on May 30, 2016)
- 5 Step Life Cycle for Neural Network Models in Keras (Jason Brownlee on August 5, 2016 in Deep Learning)

- Regression Tutorial with the Keras Deep Learning Library in Python (Jason Brownlee on June 3, 2016 in Deep Learning)
- Binary Classification Tutorial with the Keras Deep Learning Library (Jason Brownlee With Keras On Jason Brownlee on August 11, 2016 in Deep Learning)

> Keras is an extension to the popular Theano numerical library for deep learning. Keras wraps the Theano library and exposes a simple to use and extensible API for creating deep learning models. Keras is ideal for rapid experimentation with deep learning models and for those who are not expert programmers or mathematicians.
>
> Keras is a high-level API for Theano that makes it easier to define and evaluate deep learning models. It is ideal for those who are not expert programmers or mathematicians and for those who want to use the power of Theano without having to write complex numerical code.
>
> Keras is a minimalist, modular neural network library that can use either Theano or TensorFlow as its tensor manipulation library. Keras supports convolutional networks, recurrent networks, and combinations of the two. Keras supports arbitrary connectivity schemes (including multi-input and multi-output training). Keras runs seamlessly on CPU and GPU.
>
> Keras was developed and maintained by François Chollet, a Google engineer using four guiding principles:
>
> - **Modularity**: A model can be understood as a sequence or a graph alone. All the concerns of a deep learning model are discrete components that can be combined in arbitrary ways.
> - **Minimalism**: The library provides just enough to achieve an outcome. No frills, maximum readability. Ease of use.
> - **Extensibility**: New components are easy to add, and old ones are well-use within the framework, intended for researchers to try out wild new ideas.
> - **Python**: No separate model files with custom file formats. Everything is in native Python.
>
> Keras is ideal for rapid experimentation with deep learning models and for those who are not expert programmers or mathematicians.
>
> Keras is a minimalist, modular neural network library that can use either Theano or TensorFlow as its tensor manipulation library. Keras supports convolutional networks, recurrent networks, and combinations of the two. Keras supports arbitrary connectivity schemes (including multi-input and multi-output training). Keras runs seamlessly on CPU and GPU.
>
> Keras was developed and maintained by François Chollet, a Google engineer using four guiding principles:
>
> - **Modularity**: A model can be understood as a sequence or a graph alone. All the concerns of a deep learning model are discrete components that can be combined in arbitrary ways.
> - **Minimalism**: The library provides just enough to achieve an outcome. No frills, maximum readability. Ease of use.
> - **Extensibility**: New components are easy to add, and old ones are well-use within the framework, intended for researchers to try out wild new ideas.
> - **Python**: No separate model files with custom file formats. Everything is in native Python.
>
> Keras is ideal for rapid experimentation with deep learning models and for those who are not expert programmers or mathematicians.
>
> Keras is a minimalist, modular neural network library that can use either Theano or TensorFlow as its tensor manipulation library. Keras supports convolutional networks, recurrent networks, and combinations of the two. Keras supports arbitrary connectivity schemes (including multi-input and multi-output training). Keras runs seamlessly on CPU and GPU.

Keras examples — Images

- Multi-label classification with Keras (Jason Brownlee on June 7, 2016 in Deep Learning, Keras, Tutorials) — pdf



- Imagenet classification with Python and Keras by Adrian Rosebrock on August 10, 2016
- Building powerful image classification models using very little data (pdf)
- Image Recognition in Python with Keras (by Thomas Wanhamaki on 2016-08-08)

Example output of CIFAR-10 output (CPU calculations)

- Keras as a simplified interface to TensorFlow: tutorial
- Building Deep Learners on Top of Keras/Theano

> Using Keras models with TensorFlow

> RE: Keras -GPU and distributed training

> Exporting a model with TensorFlow serving

Keras — Text & Sequences

- How to use Word Embedding layers for deep learning (11 Jun 2017)

Word embedding layers are used in