

Recommended Websites on A.I./Machine Learning

Andrej Karpathy Blog

- [Hacker's guide to Neural Networks](#)
- [The Unreasonable Effectiveness of Recurrent Neural Networks](#)

i am trask

I recommend starting here for basic Neural Network concepts.

- [A Neural Network in 11 lines of Python \(Part 1\)](#)
- [A Neural Network in 13 lines of Python \(Part 2 - Gradient Descent\)](#)
- [Anyone Can Learn To Code an LSTM-RNN in Python \(Part 1: RNN\)](#)

WildML

- [Recurrent Neural Networks Tutorial, Part 1 – Introduction to RNNs](#)
- [Recurrent Neural Networks Tutorial, Part 2 – Implementing a RNN with Python, Numpy and Theano](#)
- [Recurrent Neural Networks Tutorial, Part 3 – Backpropagation Through Time and Vanishing Gradients](#)
- [Recurrent Neural Network Tutorial, Part 4 – Implementing a GRU/LSTM RNN with Python and Theano](#)

Other Sources

- [Awesome Machine Learning](#)
- [Understanding LSTM Networks](#)

Automated Fact-Checking

- [Automated Fact-Checking presentation by Joshua Chen](#)

Papers

- [Computational Fact Checking from Knowledge Networks](#)
- [Discriminative Predicate Path Mining for Fact Checking in Knowledge Graphs](#)
- [Towards Computational Fact-Checking](#)

Natural Language Processing

Copies of these papers are in the PDF directory. I recommend starting with "Text Mining: the State of the Art and the Challenges" for an overview of text mining.

General

Articles, Blogposts, and Tutorials

- [edX Course on Natural Language Processing](#)
- [Oxford Deep NLP 2017 Course](#)
- [Regular Expressions 101](#)

Papers

- [Evolving Better Stoplists for Document Clustering and Web Intelligence](#)
- [On Stopwords, Filtering and Data Sparsity for Sentiment Analysis of Twitter](#)
- [Preprocessing Techniques for Text Mining - An Overview](#)
- [Retrieval Effectiveness on the Web](#)
- [Text Mining: The State of the Art and the Challenges](#)

word2vec and doc2vec

I haven't read through these yet, but it seems that *Efficient Estimation of Word Representations in Vector Space* and *Distributed Representations of Words and Phrases and their Compositionality* started it all. Here are the links for documentation on [word2vec](#) and [doc2vec](#).

Articles, Blogposts, and Tutorials

- [A Gentle Intorduction to Doc2Vec](#)
- [Vector Representations of Words](#)
- [Word2Vec Tutorial - The Skip-Gram Model](#)
- [Word2Vec Tutorial Part 2 - Negative Sampling](#)

Papers

- [An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation](#)
- [Distributed Representations of Sentences and Documents](#)
- [Distributed Representations of Words and Phrases and their Compositionality](#)
- [Efficient Estimation of Word Representations in Vector Space](#)
- [Neural Network Doc2vec in Automated Sentiment Analysis for Short Informal Texts](#)