# 大数据分析与机器学习领域Python兵器谱

2015-1-27 17:26| 发布者: [炼数成金\_小数](http://www.dataguru.cn/space-uid-66234.html)| 查看: 23763| 评论: 0|来自: 大数据邦

**摘要**: 曾经因为NLTK的缘故开始学习Python，之后渐渐成为我工作中的第一辅助脚本语言，虽然开发语言是C/C++，但平时的很多文本数据处理任务都交给了Python。离开腾讯创业后，第一个作品课程图谱也是选择了Python系的Flask框 ...

|  |
| --- |
| 曾经因为NLTK的缘故开始学习Python，之后渐渐成为我工作中的第一辅助脚本语言，虽然开发语言是C/C++，但平时的很多文本数据处理任务都交给了Python。离开腾讯创业后，第一个作品课程图谱也是选择了Python系的Flask框架，渐渐的将自己的绝大部分工作交给了Python。这些年来，接触和使用了很多Python工具包，特别是在文本处理，科学计算，机器学习和数据挖掘领域，有很多很多优秀的Python工具包可供使用，所以作为Pythoner，也是相当幸福的。其实如果仔细留意微博，你会发现很多这方面的分享，自己也Google了一下，发现也有同学总结了“Python机器学习库”，不过总感觉缺少点什么。最近流行一个词，全栈工程师（full stack engineer），作为一个苦逼的创业者，天然的要把自己打造成一个full stack engineer，而这个过程中，这些Python工具包给自己提供了足够的火力，所以想起了这个系列。当然，这也仅仅是抛砖引玉，希望大家能提供更多的线索，来汇总整理一套Python网页爬虫，文本处理，科学计算，机器学习和数据挖掘的兵器谱。  一、Python网页爬虫工具集  一个真实的项目，一定是从获取数据开始的。无论文本处理，机器学习和数据挖掘，都需要数据，除了通过一些渠道购买或者下载的专业数据外，常常需要大家自己动手爬数据，这个时候，爬虫就显得格外重要了，幸好，Python提供了一批很不错的网页爬虫工具框架，既能爬取数据，也能获取和清洗数据，我们也就从这里开始了：  1. Scrapy  Scrapy, a fast high-level screen scraping and web crawling framework for Python.  鼎鼎大名的Scrapy，相信不少同学都有耳闻，课程图谱中的很多课程都是依靠Scrapy抓去的，这方面的介绍文章有很多，推荐大牛pluskid早年的一篇文章：《Scrapy 轻松定制网络爬虫》，历久弥新。  官方主页：http://scrapy.org/  Github代码页: https://github.com/scrapy/scrapy  2. Beautiful Soup  You didn’t write that awful page. You’re just trying to get some data out of it. Beautiful Soup is here to help. Since 2004, it’s been saving programmers hours or days of work on quick-turnaround screen scraping projects.  读书的时候通过《集体智慧编程》这本书知道Beautiful Soup的，后来也偶尔会用用，非常棒的一套工具。客观的说，Beautifu Soup不完全是一套爬虫工具，需要配合urllib使用，而是一套HTML/XML数据分析，清洗和获取工具。  官方主页：http://www.crummy.com/software/BeautifulSoup/  3. Python-Goose  Html Content / Article Extractor, web scrapping lib in Python  Goose最早是用Java写得，后来用Scala重写，是一个Scala项目。Python-Goose用Python重写，依赖了Beautiful Soup。前段时间用过，感觉很不错，给定一个文章的URL, 获取文章的标题和内容很方便。  Github主页：https://github.com/grangier/python-goose  二、Python文本处理工具集  从网页上获取文本数据之后，依据任务的不同，就需要进行基本的文本处理了，譬如对于英文来说，需要基本的tokenize，对于中文，则需要常见的中文分词，进一步的话，无论英文中文，还可以词性标注，句法分析，关键词提取，文本分类，情感分析等等。这个方面，特别是面向英文领域，有很多优秀的工具包，我们一一道来。  1. NLTK — Natural Language Toolkit  NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, and an active discussion forum.  搞自然语言处理的同学应该没有人不知道NLTK吧，这里也就不多说了。不过推荐两本书籍给刚刚接触NLTK或者需要详细了解NLTK的同学: 一个是官方的《Natural Language Processing with Python》，以介绍NLTK里的功能用法为主，同时附带一些Python知识，同时国内陈涛同学友情翻译了一个中文版，这里可以看到：推荐《用Python进行自然语言处理》中文翻译-NLTK配套书；另外一本是《Python Text Processing with NLTK 2.0 Cookbook》，这本书要深入一些，会涉及到NLTK的代码结构，同时会介绍如何定制自己的语料和模型等，相当不错。  官方主页：http://www.nltk.org/  Github代码页：https://github.com/nltk/nltk  2. Pattern  Pattern is a web mining module for the Python programming language.  It has tools for data mining (Google, Twitter and Wikipedia API, a web crawler, a HTML DOM parser), natural language processing (part-of-speech taggers, n-gram search, sentiment analysis, WordNet), machine learning (vector space model, clustering, SVM), network analysis and canvas visualization.  Pattern由比利时安特卫普大学CLiPS实验室出品，客观的说，Pattern不仅仅是一套文本处理工具，它更是一套web数据挖掘工具，囊括了数据抓取模块（包括Google, Twitter, 维基百科的API，以及爬虫和HTML分析器），文本处理模块（词性标注，情感分析等），机器学习模块(VSM, 聚类，SVM）以及可视化模块等，可以说，Pattern的这一整套逻辑也是这篇文章的组织逻辑，不过这里我们暂且把Pattern放到文本处理部分。我个人主要使用的是它的英文处理模块Pattern.en, 有很多很不错的文本处理功能，包括基础的tokenize, 词性标注，句子切分，语法检查，拼写纠错，情感分析，句法分析等，相当不错。  官方主页：http://www.clips.ua.ac.be/pattern  3. TextBlob: Simplified Text Processing  TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.  TextBlob是一个很有意思的Python文本处理工具包，它其实是基于上面两个Python工具包NLKT和Pattern做了封装（TextBlob stands on the giant shoulders of NLTK and pattern, and plays nicely with both），同时提供了很多文本处理功能的接口，包括词性标注，名词短语提取，情感分析，文本分类，拼写检查等，甚至包括翻译和语言检测，不过这个是基于Google的API的，有调用次数限制。TextBlob相对比较年轻，有兴趣的同学可以关注。  官方主页：http://textblob.readthedocs.org/en/dev/  Github代码页：https://github.com/sloria/textblob  4. MBSP for Python  MBSP is a text analysis system based on the TiMBL and MBT memory based learning applications developed at CLiPS and ILK. It provides tools for Tokenization and Sentence Splitting, Part of Speech Tagging, Chunking, Lemmatization, Relation Finding and Prepositional Phrase Attachment.  MBSP与Pattern同源，同出自比利时安特卫普大学CLiPS实验室，提供了Word Tokenization, 句子切分，词性标注，Chunking, Lemmatization，句法分析等基本的文本处理功能，感兴趣的同学可以关注。  官方主页：http://www.clips.ua.ac.be/pages/MBSP  5. Gensim: Topic modeling for humans  Gensim是一个相当专业的主题模型Python工具包，无论是代码还是文档，我们曾经用《如何计算两个文档的相似度》介绍过Gensim的安装和使用过程，这里就不多说了。  官方主页：http://radimrehurek.com/gensim/index.html  github代码页：https://github.com/piskvorky/gensim  6. langid.py: Stand-alone language identification system  语言检测是一个很有意思的话题，不过相对比较成熟，这方面的解决方案很多，也有很多不错的开源工具包，不过对于Python来说，我使用过langid这个工具包，也非常愿意推荐它。langid目前支持97种语言的检测，提供了很多易用的功能，包括可以启动一个建议的server，通过json调用其API，可定制训练自己的语言检测模型等，可以说是“麻雀虽小，五脏俱全”。  Github主页：https://github.com/saffsd/langid.py  7. Jieba: 结巴中文分词  “结巴”中文分词：做最好的Python中文分词组件 “Jieba” (Chinese for “to stutter”) Chinese text segmentation: built to be the best Python Chinese word segmentation module.  好了，终于可以说一个国内的Python文本处理工具包了：结巴分词，其功能包括支持三种分词模式（精确模式、全模式、搜索引擎模式），支持繁体分词，支持自定义词典等，是目前一个非常不错的Python中文分词解决方案。  Github主页：https://github.com/fxsjy/jieba  8. xTAS  xtas, the eXtensible Text Analysis Suite, a distributed text analysis package based on Celery and Elasticsearch.  感谢微博朋友 @大山坡的春 提供的线索：我们组同事之前发布了xTAS，也是基于python的text mining工具包，欢迎使用，链接：http://t.cn/RPbEZOW。看起来很不错的样子，回头试用一下。  Github代码页：https://github.com/NLeSC/xtas  三、Python科学计算工具包  说起科学计算，大家首先想起的是Matlab，集数值计算，可视化工具及交互于一身，不过可惜是一个商业产品。开源方面除了GNU Octave在尝试做一个类似Matlab的工具包外，Python的这几个工具包集合到一起也可以替代Matlab的相应功能：NumPy+SciPy+Matplotlib+iPython。同时，这几个工具包，特别是NumPy和SciPy，也是很多Python文本处理 & 机器学习 & 数据挖掘工具包的基础，非常重要。最后再推荐一个系列《用Python做科学计算》，将会涉及到NumPy, SciPy, Matplotlib，可以做参考。  1. NumPy  NumPy is the fundamental package for scientific computing with Python. It contains among other things:  1）a powerful N-dimensional array object  2）sophisticated (broadcasting) functions  3）tools for integrating C/C++ and Fortran code  4） useful linear algebra, Fourier transform, and random number capabilities  Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.  NumPy几乎是一个无法回避的科学计算工具包，最常用的也许是它的N维数组对象，其他还包括一些成熟的函数库，用于整合C/C++和Fortran代码的工具包，线性代数、傅里叶变换和随机数生成函数等。NumPy提供了两种基本的对象：ndarray（N-dimensional array object）和 ufunc（universal function object）。ndarray是存储单一数据类型的多维数组，而ufunc则是能够对数组进行处理的函数。  官方主页：http://www.numpy.org/  2. SciPy：Scientific Computing Tools for Python  SciPy refers to several related but distinct entities:  1）The SciPy Stack, a collection of open source software for scientific computing in Python, and particularly a specified set of core packages.  2）The community of people who use and develop this stack.  3）Several conferences dedicated to scientific computing in Python – SciPy, EuroSciPy and SciPy.in.  4）The SciPy library, one component of the SciPy stack, providing many numerical routines.  “SciPy是一个开源的Python算法库和数学工具包，SciPy包含的模块有最优化、线性代数、积分、插值、特殊函数、快速傅里叶变换、信号处理和图像处理、常微分方程求解和其他科学与工程中常用的计算。其功能与软件MATLAB、Scilab和GNU Octave类似。 Numpy和Scipy常常结合着使用，Python大多数机器学习库都依赖于这两个模块。”—-引用自“Python机器学习库”  官方主页：http://www.scipy.org/  3. Matplotlib  matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. matplotlib can be used in python scripts, the python and ipython shell (ala MATLAB®\* or Mathematica®†), web application servers, and six graphical user interface toolkits.  matplotlib 是python最著名的绘图库，它提供了一整套和matlab相似的命令API，十分适合交互式地进行制图。而且也可以方便地将它作为绘图控件，嵌入GUI应用程序中。Matplotlib可以配合ipython shell使用，提供不亚于Matlab的绘图体验，总之用过了都说好。  官方主页：http://matplotlib.org/  4. iPython  IPython provides a rich architecture for interactive computing with:  1）Powerful interactive shells (terminal and Qt-based).  2）A browser-based notebook with support for code, text, mathematical expressions, inline plots and other rich media.  3）Support for interactive data visualization and use of GUI toolkits.  4）Flexible, embeddable interpreters to load into your own projects.  5）Easy to use, high performance tools for parallel computing.  “iPython 是一个Python 的交互式Shell，比默认的Python Shell 好用得多，功能也更强大。 她支持语法高亮、自动完成、代码调试、对象自省，支持 Bash Shell 命令，内置了许多很有用的功能和函式等，非常容易使用。 ” 启动iPython的时候用这个命令“ipython –pylab”，默认开启了matploblib的绘图交互，用起来很方便。  官方主页：http://ipython.org/  四、Python 机器学习 & 数据挖掘 工具包  机器学习和数据挖掘这两个概念不太好区分，这里就放到一起了。这方面的开源Python工具包有很多，这里先从熟悉的讲起，再补充其他来源的资料，也欢迎大家补充。  1. scikit-learn: Machine Learning in Python  scikit-learn (formerly scikits.learn) is an open source machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, logistic regression, naive Bayes, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.  首先推荐大名鼎鼎的scikit-learn，scikit-learn是一个基于NumPy, SciPy, Matplotlib的开源机器学习工具包，主要涵盖分类，回归和聚类算法，例如SVM， 逻辑回归，朴素贝叶斯，随机森林，k-means等算法，代码和文档都非常不错，在许多Python项目中都有应用。例如在我们熟悉的NLTK中，分类器方面就有专门针对scikit-learn的接口，可以调用scikit-learn的分类算法以及训练数据来训练分类器模型。这里推荐一个视频，也是我早期遇到scikit-learn的时候推荐过的：推荐一个Python机器学习工具包Scikit-learn以及相关视频–Tutorial: scikit-learn – Machine Learning in Python  官方主页：http://scikit-learn.org/  2. Pandas: Python Data Analysis Library  Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.  第一次接触Pandas是由于Udacity上的一门数据分析课程“Introduction to Data Science” 的Project需要用Pandas库，所以学习了一下Pandas。Pandas也是基于NumPy和Matplotlib开发的，主要用于数据分析和数据可视化，它的数据结构DataFrame和R语言里的data.frame很像，特别是对于时间序列数据有自己的一套分析机制，非常不错。这里推荐一本书《Python for Data Analysis》，作者是Pandas的主力开发，依次介绍了iPython, NumPy, Pandas里的相关功能，数据可视化，数据清洗和加工，时间数据处理等，案例包括金融股票数据挖掘等，相当不错。  官方主页：http://pandas.pydata.org/  =======================================================  分割线，以上工具包基本上都是自己用过的，以下来源于其他同学的线索，特别是《Python机器学习库》，《23个python的机器学习包》，做了一点增删修改，欢迎大家补充  ========================================================  mlpy is a Python module for Machine Learning built on top of NumPy/SciPy and the GNU Scientific Libraries.  mlpy provides a wide range of state-of-the-art machine learning methods for supervised and unsupervised problems and it is aimed at finding a reasonable compromise among modularity, maintainability, reproducibility, usability and efficiency. mlpy is multiplatform, it works with Python 2 and 3 and it is Open Source, distributed under the GNU General Public License version 3.  官方主页：http://mlpy.sourceforge.net/  4. MDP：The Modular toolkit for Data Processing  Modular toolkit for Data Processing (MDP) is a Python data processing framework.  From the user’s perspective, MDP is a collection of supervised and unsupervised learning algorithms and other data processing units that can be combined into data processing sequences and more complex feed-forward network architectures.  From the scientific developer’s perspective, MDP is a modular framework, which can easily be expanded. The implementation of new algorithms is easy and intuitive. The new implemented units are then automatically integrated with the rest of the library.  The base of available algorithms is steadily increasing and includes signal processing methods (Principal Component Analysis, Independent Component Analysis, Slow Feature Analysis), manifold learning methods ([Hessian] Locally Linear Embedding), several classifiers, probabilistic methods (Factor Analysis, RBM), data pre-processing methods, and many others.  “MDP用于数据处理的模块化工具包，一个Python数据处理框架。 从用户的观点，MDP是能够被整合到数据处理序列和更复杂的前馈网络结构的一批监督学习和非监督学习算法和其他数据处理单元。计算依照速度和内存需求而高效的执行。从科学开发者的观点，MDP是一个模块框架，它能够被容易地扩展。新算法的实现是容易且直观的。新实现的单元然后被自动地与程序库的其余部件进行整合。MDP在神经科学的理论研究背景下被编写，但是它已经被设计为在使用可训练数据处理算法的任何情况中都是有用的。其站在用户一边的简单性，各种不同的随时可用的算法，及应用单元的可重用性，使得它也是一个有用的教学工具。”  官方主页：http://mdp-toolkit.sourceforge.net/  5. PyBrain  PyBrain is a modular Machine Learning Library for Python. Its goal is to offer flexible, easy-to-use yet still powerful algorithms for Machine Learning Tasks and a variety of predefined environments to test and compare your algorithms.  PyBrain is short for Python-Based Reinforcement Learning, Artificial Intelligence and Neural Network Library. In fact, we came up with the name first and later reverse-engineered this quite descriptive “Backronym”.  “PyBrain(Python-Based Reinforcement Learning, Artificial Intelligence and Neural Network)是Python的一个机器学习模块，它的目标是为机器学习任务提供灵活、易应、强大的机器学习算法。（这名字很霸气）  PyBrain正如其名，包括神经网络、强化学习(及二者结合)、无监督学习、进化算法。因为目前的许多问题需要处理连续态和行为空间，必须使用函数逼近(如神经网络)以应对高维数据。PyBrain以神经网络为核心，所有的训练方法都以神经网络为一个实例。”  官方主页：http://www.pybrain.org/  6. PyML – machine learning in Python  PyML is an interactive object oriented framework for machine learning written in Python. PyML focuses on SVMs and other kernel methods. It is supported on Linux and Mac OS X.  “PyML是一个Python机器学习工具包，为各分类和回归方法提供灵活的架构。它主要提供特征选择、模型选择、组合分类器、分类评估等功能。”  项目主页：http://pyml.sourceforge.net/  7. Milk：Machine learning toolkit in Python.  Its focus is on supervised classification with several classifiers available:  SVMs (based on libsvm), k-NN, random forests, decision trees. It also performs  feature selection. These classifiers can be combined in many ways to form  different classification systems.  “Milk是Python的一个机器学习工具箱，其重点是提供监督分类法与几种有效的分类分析：SVMs(基于libsvm)，K-NN，随机森林经济和决策树。它还可以进行特征选择。这些分类可以在许多方面相结合，形成不同的分类系统。对于无监督学习，它提供K-means和affinity propagation聚类算法。”  官方主页：http://luispedro.org/software/milk  http://luispedro.org/software/milk  8. PyMVPA: MultiVariate Pattern Analysis (MVPA) in Python  PyMVPA is a Python package intended to ease statistical learning analyses of large datasets. It offers an extensible framework with a high-level interface to a broad range of algorithms for classification, regression, feature selection, data import and export. It is designed to integrate well with related software packages, such as scikit-learn, and MDP. While it is not limited to the neuroimaging domain, it is eminently suited for such datasets. PyMVPA is free software and requires nothing but free-software to run.  “PyMVPA(Multivariate Pattern Analysis in Python)是为大数据集提供统计学习分析的Python工具包，它提供了一个灵活可扩展的框架。它提供的功能有分类、回归、特征选择、数据导入导出、可视化等”  官方主页：http://www.pymvpa.org/  9. Pyrallel – Parallel Data Analytics in Python  Experimental project to investigate distributed computation patterns for machine learning and other semi-interactive data analytics tasks.  “Pyrallel(Parallel Data Analytics in Python)基于分布式计算模式的机器学习和半交互式的试验项目，可在小型集群上运行”  Github代码页：http://github.com/pydata/pyrallel  10. Monte – gradient based learning in Python  Monte (python) is a Python framework for building gradient based learning machines, like neural networks, conditional random fields, logistic regression, etc. Monte contains modules (that hold parameters, a cost-function and a gradient-function) and trainers (that can adapt a module’s parameters by minimizing its cost-function on training data).  Modules are usually composed of other modules, which can in turn contain other modules, etc. Gradients of decomposable systems like these can be computed with back-propagation.  “Monte (machine learning in pure Python)是一个纯Python机器学习库。它可以迅速构建神经网络、条件随机场、逻辑回归等模型，使用inline-C优化，极易使用和扩展。”  官方主页：http://montepython.sourceforge.net  11. Theano  Theano is a Python library that allows you to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays efficiently. Theano features:  1）tight integration with NumPy – Use numpy.ndarray in Theano-compiled functions.  2）transparent use of a GPU – Perform data-intensive calculations up to 140x faster than with CPU.(float32 only)  3）efficient symbolic differentiation – Theano does your derivatives for function with one or many inputs.  4）speed and stability optimizations – Get the right answer for log(1+x) even when x is really tiny.  5）dynamic C code generation – Evaluate expressions faster.  6） extensive unit-testing and self-verification – Detect and diagnose many types of mistake.  Theano has been powering large-scale computationally intensive scientific investigations since 2007. But it is also approachable enough to be used in the classroom (IFT6266 at the University of Montreal).  “Theano 是一个 Python 库，用来定义、优化和模拟数学表达式计算，用于高效的解决多维数组的计算问题。Theano的特点：紧密集成Numpy；高效的数据密集型GPU计算；高效的符号微分运算；高速和稳定的优化；动态生成c代码；广泛的单元测试和自我验证。自2007年以来，Theano已被广泛应用于科学运算。theano使得构建深度学习模型更加容易，可以快速实现多种模型。PS：Theano，一位希腊美女，Croton最有权势的Milo的女儿，后来成为了毕达哥拉斯的老婆。”  12. Pylearn2  Pylearn2 is a machine learning library. Most of its functionality is built on top of Theano. This means you can write Pylearn2 plugins (new models, algorithms, etc) using mathematical expressions, and theano will optimize and stabilize those expressions for you, and compile them to a backend of your choice (CPU or GPU).  “Pylearn2建立在theano上，部分依赖scikit-learn上，目前Pylearn2正处于开发中，将可以处理向量、图像、视频等数据，提供MLP、RBM、SDA等深度学习模型。”  官方主页：http://deeplearning.net/software/pylearn2/ |