[常用第三方开源代码库 （thirdparty/common）](http://blog.csdn.net/u013005493/article/details/17044075)

2013-12-01 12:14 2997人阅读 [评论](http://blog.csdn.net/u013005493/article/details/17044075#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/u013005493/article/details/17044075#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

Go 分布式系统（2） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg 数据存储（2） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg 源码剖析 Python C++（1） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg机器学习（3） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg 计算广告学（2） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg 自然语言处理（1） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为博主原创文章，未经博主允许不得转载。

目录[(?)[+]](http://blog.csdn.net/u013005493/article/details/17044075)

在工作和学习中，借助第三方开源代码库是常见的事情，“站在巨人的肩膀上”嘛，相信大家都不会陌生，赞叹开源、共享的伟大。

一方面为了做个总结，另一方面，就是好东西要与大家分享，我在 Github 上维护了一个页面 <https://github.com/fandywang/thirdparty_intro>，包含了个人比较关注的第三方代码库，如下（持续更新中）：

**Google 开源库**

* [zh-google-styleguide](http://zh-google-styleguide.readthedocs.org/en/latest/) - Google 开源项目风格指南.
* [protobuf](https://code.google.com/p/protobuf/) - Protocol Buffers - Google's data interchange format.
* [gflags](https://code.google.com/p/gflags/) - Commandline flags module for C++.
* [glog](https://code.google.com/p/google-glog/) - Logging library for C++.
* [gtest](https://code.google.com/p/googletest/) - Google C++ Testing Framework.
* [googlemock](https://code.google.com/p/googlemock/) - Google C++ Mocking Framework.
* [leveldb](http://code.google.com/p/leveldb/) - A fast and lightweight key/value database library by Google. [cpy-leveldb](https://github.com/carlopires/cpy-leveldb) - Python bindings for LevelDB using leveldb c api.
* [The Chromium Projects](http://www.chromium.org/) - The Chromium projects include Chromium and Chromium OS, the open-source projects behind the Google Chrome browser and Google Chrome OS, respectively.

**C++ base 库**

* [toft](https://github.com/chen3feng/toft) - C++ Base Library for Linux server side development. [thirdparty](https://github.com/chen3feng/thirdparty) - Put thirdparty library here for toft ant foxy. [chen3feng](https://github.com/chen3feng)
* [folly](https://github.com/facebook/folly) - Folly is an open-source C++ library developed and used at Facebook.

**算法和数据结构**

* [darts-clone](https://code.google.com/p/darts-clone/) - A clone of the Darts (Double-ARray Trie System).
* [Darts](http://chasen.org/~taku/software/darts/) - Double-ARray Trie System. [中文翻译文档](http://www.52nlp.cn/darts-double-array-trie-system-%E7%BF%BB%E8%AF%91%E6%96%87%E6%A1%A3)
* [sparsehash](https://code.google.com/p/sparsehash/?redir=1) - An extremely memory-efficient hash\_map implementation。
* [cityhash](https://code.google.com/p/cityhash/) - The CityHash family of hash functions.
* [stringencoders](https://code.google.com/p/stringencoders/) - A collection of high performance c-string transformations, frequently 2x faster than standard implementations (if they exist at all).
* [Numpy](http://www.numpy.org/) - NumPy is the fundamental package for scientific computing with Python.

**自然语言处理库**

* [NLTK](https://github.com/nltk/nltk) - NLTK -- the Natural Language Toolkit -- is a suite of open source Python modules, data sets and tutorials supporting research and development in Natural Language Processing. [NLTK Book](https://github.com/nltk/nltk_book)
* [jieba](https://github.com/fxsjy/jieba) - 结巴中文分词.
* [gensim](https://github.com/piskvorky/gensim) - Gensim is a Python library for topic modelling, document indexing and similarity retrieval with large corpora. Target audience is the natural language processing (NLP) and information retrieval (IR) community.
* [LTP](http://www.ltp-cloud.com/) - 语言技术平台（Language Technology Platform，LTP）是哈工大社会计算与信息检索研究中心历时十年研制的一整套开放中文自然语言处理系统。
* [Stanford CoreNLP](http://nlp.stanford.edu/software/corenlp.shtml) - Stanford CoreNLP provides a set of natural language analysis tools which can take raw English language text input and give the base forms of words, their parts of speech, whether they are names of companies, people, etc., normalize dates, times, and numeric quantities, and mark up the structure of sentences in terms of phrases and word dependencies, and indicate which noun phrases refer to the same entities.
* [openNLP](http://opennlp.apache.org/) - The Apache OpenNLP library is a machine learning based toolkit for the processing of natural language text.
* [SRILM](http://www.speech.sri.com/projects/srilm/) - SRILM is a toolkit for building and applying statistical language models (LMs), primarily for use in speech recognition, statistical tagging and segmentation, and machine translation.
* [IRSTLM](http://hlt.fbk.eu/en/irstlm) - The IRST Language Modeling Toolkit features algorithms and data structures suitable to estimate, store, and access very large LMs.
* [KenLM](http://kheafield.com/code/kenlm/estimation/) - KenLM estimates unpruned language models with modified Kneser-Ney smoothing.
* [Moses](http://www.statmt.org/moses/) - Moses is a statistical machine translation system that allows you to automatically train translation models for any language pair.
* [GIZA++](https://code.google.com/p/giza-pp/) - GIZA++ is a statical machine translation toolkit that is used to train IBM Models 1-5 and an HMM word alignment model.
* [genius](https://github.com/duanhongyi/genius) - genius中文分词，是基于crf条件随机场的分组件.
* [sego](https://github.com/huichen/sego) - Go中文分词.
* [pinyin](https://github.com/huichen/pinyin) - Go语言汉字转拼音工具.
* [ReVerb](https://github.com/knowitall/reverb/) - ReVerb is a program that automatically identifies and extracts binary relationships from English sentences. ReVerb is designed for Web-scale information extraction, where the target relations cannot be specified in advance and speed is important.
* [Statistical natural language processing and corpus-based computational linguistics: An annotated list of resources](http://www-nlp.stanford.edu/links/statnlp.html) - 斯坦福自然语言组的NLP及计算语言学的资料汇总：包括各种工具，代码，语料库，字典，课程的链接及简单介绍。<http://t.cn/zOfVAzs>

**信息检索库**

* [Lemur](http://www.lemurproject.org/) - The Lemur Project develops search engines, browser toolbars, text analysis tools, and data resources that support research and development of information retrieval and text mining software.
* [Lucene](http://lucene.apache.org/) - The Apache Lucene project develops open-source search software.
* [gensim](https://github.com/piskvorky/gensim) - Gensim is a Python library for topic modelling, document indexing and similarity retrieval with large corpora. Target audience is the natural language processing (NLP) and information retrieval (IR) community.
* [wukong](https://github.com/huichen/wukong) - 悟空全文搜索引擎.
* [Scrapy](http://scrapy.org/) - a fast high-level screen scraping and web crawling framework for Python.
* [distribute\_crawler](https://github.com/gnemoug/distribute_crawler) - 使用scrapy,redis, mongodb,graphite实现的一个分布式网络爬虫,底层存储mongodb集群,分布式使用redis实现, 爬虫状态显示使用graphite实现。

**机器学习库**

* [LASSO](https://github.com/wangkuiyi/lasso) - LASSO is a parallel machine learning system that learns a regression model from large data. It works in either of two modes: IPM-mode and MPI-mode.
* [libsvm](http://www.csie.ntu.edu.tw/~cjlin/libsvm/) - A Library for Support Vector Machines. [支持向量机通俗导论（理解SVM的三层境界）](http://blog.csdn.net/v_july_v/article/details/7624837) 来自[研究者July](http://weibo.com/julyweibo?s=6cm7D0). 在本文中，你将看到，理解SVM分三层境界， 第一层: 了解SVM(你只需要对SVM有个大致的了解，知道它是个什么东西便已足够)； 第二层: 深入SVM(你将跟我一起深入SVM的内部原理，通晓其各处脉络，以为将来运用它时游刃有余)； 第三层: 证明SVM(当你了解了所有的原理之后，你会有大笔一挥，尝试证明它的冲动)。
* [liblinear](http://www.csie.ntu.edu.tw/~cjlin/liblinear/) - A Library for Large Linear Classification.
* [RankLib](http://people.cs.umass.edu/~vdang/ranklib.html) - RankLib is a library of learning to rank algorithms.
* [svmlight](http://www.cs.cornell.edu/people/tj/svm_light/) - SVMlight is an implementation of Support Vector Machines (SVMs) in C.
* [plda](http://code.google.com/p/plda/) - A parallel C++ implementation of fast Gibbs sampling of Latent Dirichlet Allocation
* [GibbsLDA++](http://gibbslda.sourceforge.net/) - A C/C++ implementation of Latent Dirichlet Allocation (LDA) using Gibbs Sampling technique for parameter estimation and inference.
* [Yahoo\_LDA](https://github.com/shravanmn/Yahoo_LDA) - Yahoo!'s topic modelling framework using Latent Dirichlet Allocation
* [word2vec](https://code.google.com/p/word2vec/) - Tool for computing continuous distributed representations of words. [Parallelizing word2vec in Python](http://radimrehurek.com/2013/10/parallelizing-word2vec-in-python/)
* [Maximum Entropy Modeling Toolkit for Python and C++](https://github.com/lzhang10/maxent) - This package provides a (Conditional) Maximum Entropy Modeling Toolkit for Python and C++.
* [maxent](http://www.nactem.ac.uk/tsuruoka/maxent/) - A simple C++ library for maximum entropy classification.
* [easyME](https://github.com/nicyun/easyME) - This is a simple implementation of Maximum Entropy model. Algorithms implemented include: GIS, SCGIS, LBFGS, Gaussian smoothing and Exponential smoothing.
* [libLBFGS](http://www.chokkan.org/software/liblbfgs/) - This library is a C port of the implementation of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method written by Jorge Nocedal.
* [OWL-QN](http://research.microsoft.com/en-us/downloads/b1eb1016-1738-4bd5-83a9-370c9d498a03/) - The Orthant-Wise Limited-memory Quasi-Newton algorithm (OWL-QN) is a numerical optimization procedure for finding the optimum of an objective of the form {smooth function} plus {L1-norm of the parameters}. It has been used for training log-linear models (such as logistic regression) with L1-regularization.
* [CRF++](http://crfpp.googlecode.com/svn/trunk/doc/index.html?source=navbar) - CRF++ is a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting/labeling sequential data. CRF++ is designed for generic purpose and will be applied to a variety of NLP tasks, such as Named Entity Recognition, Information Extraction and Text Chunking.
* [CRFsuite](http://www.chokkan.org/software/crfsuite/) - A fast implementation of Conditional Random Fields (CRFs).
* [Wapiti](http://wapiti.limsi.fr/) - Wapiti is a very fast toolkit for segmenting and labeling sequences with discriminative models. It is based on maxent models, maximum entropy Markov models and linear-chain CRF and proposes various optimization and regularization methods to improve both the computational complexity and the prediction performance of standard models.
* [sofia-ml](http://code.google.com/p/sofia-ml/) - Suite of Fast Incremental Algorithms for Machine Learning. Includes methods for learning classification and ranking models, using Pegasos SVM, SGD-SVM, ROMMA, Passive-Aggressive Perceptron, Perceptron with Margins, and Logistic Regression.
* [mahout](http://mahout.apache.org/) - The Apache Mahout machine learning library's goal is to build scalable machine learning libraries.
* [MLTK](https://github.com/fandywang/mltk) - MLTK -- the Machine Learning Toolkit -- is a suite of C++ open source modules of Machine Learning.
* [FP-growth](https://github.com/enaeseth/python-fp-growth) - An implementation of the FP-growth algorithm in pure Python.
* [MLcomp](http://mlcomp.org/) - MLcomp is a free website for objectively comparing machine learning programs across various datasets for multiple problem domains.

**数据交换协议**

* [protobuf](https://code.google.com/p/protobuf/) - Protocol Buffers - Google's data interchange format.
* [jsoncpp](http://jsoncpp.sourceforge.net/) - JSON data format manipulation library.
* [tinyxml2](http://www.grinninglizard.com/tinyxml2/index.html) - TinyXML-2 is a simple, small, efficient, C++ XML parser that can be easily integrating into other programs.
* [thrift](http://thrift.apache.org/) - The Apache Thrift software framework, for scalable cross-language services development, combines a software stack with a code generation engine to build services that work efficiently and seamlessly between C++, Java, Python, PHP, Ruby, Erlang, Perl, Haskell, C#, Cocoa, JavaScript, Node.js, Smalltalk, OCaml and Delphi and other languages.

**数据库**

* [MySQL++](http://tangentsoft.net/mysql++/) - MySQL++ is a C++ wrapper for MySQL’s C API.
* [MongodDB](http://www.mongodb.org/) - MongoDB (from "humongous") is an open-source document database, and the leading NoSQL database. Written in C++.
* [memcached](http://memcached.org/) - Free & open source, high-performance, distributed memory object caching system, generic in nature, but intended for use in speeding up dynamic web applications by alleviating database load.
* [leveldb](http://code.google.com/p/leveldb/) - A fast and lightweight key/value database library by Google.
* [SSDB](https://github.com/ideawu/ssdb) - A fast NoSQL database server with zset data type, an alternative to Redis. SSDB is a high performace key-value(key-string, key-zset, key-hashmap) NoSQL persistent storage server, using Google LevelDB as storage engine. SSDB is stable, production-ready and is widely used by many Internet companies such as QIHU 360.
* [RocksDB](http://rocksdb.org/index.html) - RocksDB is an embeddable persistent key-value store for fast storage. RocksDB can also be the foundation for a client-server database but our current focus is on embedded workloads. RocksDB builds on LevelDB to be scalable to run on servers with many CPU cores, to efficiently use fast storage, to support IO-bound, in-memory and write-once workloads, and to be flexible to allow for innovation.
* [fatcache](https://github.com/twitter/fatcache) - Memcache on SSD. Think of fatcache as a cache for your big data.
* [THUIRDB](http://xunren.thuir.org/thuirdb/) - THUIRDB是一个C++语言实现的基础库，用于在单机上实现高性能key-value持久化存储和高速查询。[THUIRDB Paper](http://www.jofcis.com/publishedpapers/2013_9_6_2347_2355.pdf)

**网络编程**

* [thrift](http://thrift.apache.org/) - The Apache Thrift software framework, for scalable cross-language services development, combines a software stack with a code generation engine to build services that work efficiently and seamlessly between C++, Java, Python, PHP, Ruby, Erlang, Perl, Haskell, C#, Cocoa, JavaScript, Node.js, Smalltalk, OCaml and Delphi and other languages.
* [server1](https://code.google.com/p/server1/) - a c++ network server/client framework.
* [muduo-protorpc](https://github.com/chenshuo/muduo-protorpc) - Google Prorobuf RPC based on Muduo.

**Web 开发**

* [Flask](http://flask.pocoo.org/) - Flask is a microframework for Python based on Werkzeug and Jinja2. It's intended for getting started very quickly and was developed with best intentions in mind. [中文docs](https://github.com/dormouse/Flask_Docs_ZhCn)
* [Bootstrap](http://getbootstrap.com/) - Sleek, intuitive, and powerful front-end framework for faster and easier web development.
* [Django](https://www.djangoproject.com/) - Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

**分布式计算**

* [Hadoop](http://hadoop.apache.org/) - The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models.
* [ZooKeeper](http://zookeeper.apache.org/) - ZooKeeper is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services.
* [Storm](http://storm-project.net/) - Distributed and fault-tolerant realtime computation. [Storm 维基](https://github.com/nathanmarz/storm/wiki) - 提供了有关 Storm、它的理论基础的大量优秀文档，以及有关获取 Storm 和设置新项目的各种教程。您还将找到一些有关 Storm 的许多方面的实用文档，包括 Storm 在本地模式、集群模式和在 Amazon 上的使用。 GitHub 上提供了 Storm 的一个 [thorough class tree exists](http://nathanmarz.github.io/storm/doc/index.html)，详细介绍了 Storm 的类和接口。 [使用 Twitter Storm 处理实时的大数据](http://www.ibm.com/developerworks/cn/opensource/os-twitterstorm/) - 流式处理大数据简介 简介： Storm 是一个开源的、大数据处理系统，与其他系统不同，它旨在用于分布式实时处理且与语言无关。了解 Twitter Storm、它的架构，以及批处理和流式处理解决方案的发展形势。[Storm 入门教程](http://blog.linezing.com/category/storm-quick-start?spm=0.0.0.0.oU9c5c) - 来自[量子恒道官方博客](http://blog.linezing.com/?spm=0.0.0.0.5QdtJI) [storm-starter](https://github.com/nathanmarz/storm-starter) - Learn to use Storm!
* [Spark](http://spark.incubator.apache.org/) - Lightning-Fast Cluster Computing.
* [Puppet](http://puppetlabs.com/puppet/what-is-puppet) - Puppet is IT automation software that helps system administrators manage infrastructure throughout its lifecycle, from provisioning and configuration to orchestration and reporting. Using Puppet, you can easily automate repetitive tasks, quickly deploy critical applications, and proactively manage change, scaling from 10s of servers to 1000s, on-premise or in the cloud.
* [Skynet](https://github.com/skynetservices/skynet) - Skynet is a framework for distributed services in Go.
* [Kafka](http://kafka.apache.org/) - 分布式消息队列系统，A high-throughput distributed messaging system. [Kafka paper: Building LinkedIn’s Real-time Activity Data Pipeline](http://sites.computer.org/debull/A12june/pipeline.pdf) [Kafka Clients](https://cwiki.apache.org/confluence/display/KAFKA/Clients)  
  [librdkafka](https://github.com/edenhill/librdkafka) [kafka-python](https://github.com/mumrah/kafka-python)
* [Celery --- Distributed Task Queue](https://github.com/ghosert/VimProject/tree/master/StudyPyramid/celery) - 这个框架几乎是 Python 下异步消息架构的终极解决方案.
* [mapreduce-lite](https://github.com/wangkuiyi/mapreduce-lite) - A C++ implementaton of MapReduce without distributed filesystem.
* [GraphChi](http://graphlab.org/graphchi/) - GraphChi[huahua] is a spin-off of the GraphLab[rador's retriever] project. GraphChi can run very large graph computations on just a single machine, by using a novel algorithm for processing the graph from disk (SSD or hard drive). Programs for GraphChi are written in similar vertex-centric model as GraphLab. GraphChi runs vertex-centric programs asynchronously (i.e changes written to edges are immediately visible to subsequent computation), and in parallel. GraphChi also supports streaming graph updates and changing the graph structure while computing. [GraphChi ppt](http://t.cn/8DDPr7N). [GraphChi Paper](http://select.cs.cmu.edu/publications/paperdir/osdi2012-kyrola-blelloch-guestrin.pdf). [GraphChi Video](https://www.usenix.org/conference/osdi12/126-graphchi-large-scale-graph-computation-just-pc). [GraphChi's C++ version.](https://github.com/GraphChi/graphchi-cpp)-disk-based large-scale graph computation. Big Data - small machine.
* [Giraph](http://giraph.apache.org/) - Large-scale graph processing on Hadoop.
* [Celery --- Distributed Task Queue](https://github.com/ghosert/VimProject/tree/master/StudyPyramid/celery) - Celery is a simple, flexible and reliable distributed system to process vast amounts of messages, while providing operations with the tools required to maintain such a system. It’s a task queue with focus on real-time processing, while also supporting task scheduling. 这个框架几乎是 Python 下异步消息架构的终极解决方案.

**正则表达式**

* [re2](http://code.google.com/p/re2/) - an efficient, principled regular expression library.

**编译工具**

* [SCons](http://www.scons.org/) - SCons is an Open Source software construction tool—that is, a next-generation build tool. Think of SCons as an improved, cross-platform substitute for the classic Make utility with integrated functionality similar to autoconf/automake and compiler caches such as ccache. In short, SCons is an easier, more reliable and faster way to build software.
* [CMake](http://www.cmake.org/) - the cross-platform, open-source build system.
* [blade](http://code.tencent.com/projects/blade/) - Blade is designed to be a modernize building system. [Mac OS X port of Typhoon Blade](https://github.com/wangkuiyi/typhoon-blade)
* [bobo](https://github.com/bin3/bobo) - Bobo is an easy to use building tool inspired by blade.

**Code Review**

* [rietveld](http://code.google.com/p/rietveld/) - Code Review, hosted on Google App Engine.
* [Review Board](http://www.reviewboard.org/) - Take the pain out of code review.

**vim**

* [spf13-vim](http://vim.spf13.com/) - spf13-vim is a distribution of vim plugins and resources for Vim, GVim and MacVim. It is a completely cross platform distribution that stays true to the feel of vim while providing modern features like a plugin management system, autocomplete, tags and tons more.
* [Maximum Awesome](https://github.com/square/maximum-awesome) - Config files for vim and tmux, lovingly tended by a small subculture of peace-loving hippies. Built for Mac OS X.
* [VimClojure](https://github.com/vim-scripts/VimClojure) - A filetype, syntax and indent plugin for Clojure.

**Go 学习**

* [glog](https://github.com/golang/glog) - Leveled execution logs for Go.
* [groupcache](https://github.com/golang/groupcache) - groupcache is a caching and cache-filling library, intended as a replacement for memcached in many cases.
* [go-slab](https://github.com/couchbaselabs/go-slab) - A slab allocator library in the Go Programming Language.
* [Go语言资料收集](https://github.com/wonderfo/wonderfogo/wiki) -

**Python 学习**

* [pycrumbs](https://github.com/kirang89/pycrumbs) - Bits and Bytes of Python from the Internet.

**自动化部署引擎**

* [Docker](http://www.docker.io/) - Docker is an open-source project to easily create lightweight, portable, self-sufficient containers from any application. The same container that a developer builds and tests on a laptop can run at scale, in production, on VMs, bare metal, OpenStack clusters, public clouds and more. Docker 是一个开源自动化部署引擎，它可以将任何应用封装成一个简单、便携、不依赖于其他组件的容器，从而轻松地将其部署在各种虚拟环境中，以便进行各种调试。它既保证了应用的私有性，同时缩短了调试部署的周期，使得测试-封装-部署变得更加容易和便捷。不过现在Docker还在加紧开发中，相信等它开发完毕后，它会给人们的开发带来前所未有的便捷。

**其他**

* [Valgrind](http://valgrind.org/) - Valgrind is an instrumentation framework for building dynamic analysis tools. There are Valgrind tools that can automatically detect many memory management and threading bugs, and profile your programs in detail. You can also use Valgrind to build new tools.

P.S.: 俗话说，“知其然，更要知其所以然”，所以，使用开源库的同时，最好能了解其原理，可以通过阅读源代码和文档获取，当然，如果能够自己实现下最好了。