

bbking

My angel love sunshine !

Wiki语料处理

最近在做知识图谱相关工作，源数据主要来自百度百科，互动百科，中文维基百科等。其中中文维基百科提供数据库下载，下文主要讨论如何处理Wiki数据。

1. 中文维基数据下载

下载dump： <https://dumps.wikimedia.org/zhwiki/latest/>，维基数据主要包含以下几部分

zhwiki-latest-pages-articles.xml.bz2	词条正文
zhwiki-latest-redirect.sql	词条重定向（同义词）
zhwiki-latest-pagelinks.sql	词条页面内容外链
zhwiki-latest-page.sql	词条标题及摘要
zhwiki-latest-categorylinks.sql	词条开放分类链接

本文处理的数据是：[zhwiki-latest-pages-articles.xml.bz2](#)

2. 数据的抽取

Gensim是一个相当专业的主题模型Python工具包，提供了wiki数据的抽取处理类[WikiCorpus](#)，能对下载的数据（*articles.xml.bz2）进行抽取处理，得到纯净的文本语料。



公告

中山大学
信息科学与技术学院
本科
新浪微博：[@中山大学BBking](#)

昵称：[bbking](#)
园龄：[4年7个月](#)
粉丝：[51](#)
关注：[9](#)
[+加关注](#)

导航

[博客园](#)
[新随笔](#)
[联系](#)
[管理](#)


随笔分类(50)

[Cocos2d-x\(1\)](#)
[Linux\(11\)](#)
[PHP\(4\)](#)
[Python\(12\)](#)
[机器学习\(1\)](#)

```
class WikiCorpus(TextCorpus):
    """
    Treat a wikipedia articles dump (*articles.xml.bz2) as a (read-only) corpus.
    The documents are extracted on-the-fly, so that the whole (massive) dump
    can stay compressed on disk.
    >>> wiki = WikiCorpus('enwiki-20100622-pages-articles.xml.bz2') # create word-
>word_id mapping, takes almost 8h
    >>> MmCorpus.serialize('wiki_en_vocab200k.mm', wiki) # another 8h, creates a file in
MatrixMarket format plus file with id->word
    """
```



源码在此，感兴趣的可以详细品味。下面是处理代码 process_wiki_1.py，将wiki数据处理得到文本语料 wiki.zh.txt，860M。

 process_wiki_1.py

3. 数据预处理

由于中文维基包含繁体字及不规范字符，需要进行繁体转简体，以及字符编码转换。同时为了后续工作，需要对语料进行分词处理。

(1) 繁体转简体：使用的是开源简繁转换工具 [OpenCC](#)，安装说明在此，下面是linux下安装方式。

```
sudo apt-get install opencc
```

(2) 字符编码转换：使用iconv命令将文件转换成utf-8编码

```
iconv -c -t UTF-8 < input_file > output_file
#iconv -c -t UTF-8 input_file -o output_file
```

(3) 分词处理：使用 [jieba](#) 分词工具包，命令行分词

```
python -m jieba input_file > cut_file
```

[数据结构\(14\)](#)

[自然语言处理\(7\)](#)

积分与排名

积分 - 65592

排名 - 5009

阅读排行榜

1. [Python TF-IDF计算100份文档关键词权重\(27987\)](#)

2. [Python 利用pytesseract模块识别图像文字\(24428\)](#)

3. [PHP 调用Python脚本\(15670\)](#)

4. [Python 结巴分词\(9495\)](#)

5. [Python 主成分分析PCA\(9435\)](#)

评论排行榜

1. [Python TF-IDF计算100份文档关键词权重\(16\)](#)

2. [Python 手写数字识别-knn算法应用\(8\)](#)

3. [word2vec + transE 知识表示模型\(4\)](#)

4. [Gensim LDA主题模型实验\(4\)](#)

5. [Wiki语料处理\(4\)](#)

推荐排行榜

1. [Python 手写数字识别-knn算法应用\(4\)](#)

2. [word2vec + transE 知识表示模型\(3\)](#)

3. [CNN for NLP \(CS224D\)\(3\)](#)

4. [Python 利用pytesseract模块识别图像文字\(3\)](#)

5. [PHP 调用Python脚本\(2\)](#)

下面是处理代码 process_wiki_2.sh

```
+ process_wiki_2.sh
```

4. 实验结果

处理器 Intel(R) Xeon(R) CPU X5650 @ 2.67GHz

数据处理过程：主要是分词耗时48m4s。



```
openc: Traditional Chinese to Simplified Chinese...
```

```
real    0m57.765s
```

```
user    0m45.494s
```

```
sys     0m6.910s
```

```
-----
```

```
jieba: Cut words...
```

```
Building prefix dict from /usr/local/lib/python2.7/dist-packages/jieba/dict.txt ...
```

```
Loading model from cache /tmp/jieba.cache
```

```
Dumping model to file cache /tmp/jieba.cache
```

```
Loading model cost 2.141 seconds.
```

```
Prefix dict has been built successfully.
```

```
real    48m4.259s
```

```
user    47m36.987s
```

```
sys     0m22.746s
```

```
-----
```

```
iconv: ascii to utf-8...
```

```
real    0m22.039s
```

```
user    0m9.304s
```

```
sys     0m3.464s
```



数据处理结果：1.1G 已分词的中文语料

```
-rw-r--r-- 1 chenbingjin data 860M 7月 2 14:33 wiki.zh.txt
-rw-r--r-- 1 chenbingjin data 860M 7月 2 17:46 wiki.zh.chs.txt
-rw-r--r-- 1 chenbingjin data 1.1G 7月 2 18:34 wiki.zh.seg.txt
-rw-r--r-- 1 chenbingjin data 1.1G 7月 2 18:34 wiki.zh.seg.utf.txt
```

补充：[未分词的wiki语料](#)，有需要的朋友可以下载

参考

1. licstar的博客：[维基百科简体中文语料的获取](#)
2. 52nlp：[中英文维基百科语料上的word2vec实验](#)

分类: [自然语言处理](#)

标签: [wiki](#), [gensim](#)

好文要顶

关注我

收藏该文



bbking

关注 - 9

粉丝 - 51

[+加关注](#)

1

0

« 上一篇：[GPU 加速NLP任务 \(Theano+CUDA\)](#)

» 下一篇：[Gensim LDA主题模型实验](#)

posted on 2016-07-02 21:22 [bbking](#) 阅读(5243) 评论(4) [编辑](#) [收藏](#)

评论

#1楼 2016-07-03 10:41 [深度客](#)

请问楼主百度百科的数据从哪里获取呢？

[支持\(0\)](#) [反对\(0\)](#)

#2楼[楼主] 2016-07-03 11:06 [bbking](#)

@ 深度客

百度百科数据只能通过id爬取，数据是最乱但最多的

[支持\(0\)](#) [反对\(0\)](#)

#3楼 2016-07-06 06:52 [深度客](#)

@ bbking

写了一个爬虫，但感觉单机速度太慢了，跑了两天才20万条，因为研究需要，主要抓取了摘要和开放分类信息。不知楼主总共爬了多少条记录，看了下百度自己的描述，大概总共有1300多万的词条，不知可否共享下数据？

[支持\(0\)](#) [反对\(0\)](#)

#4楼[楼主] 2016-07-06 10:16 [bbking](#)

@ 深度客

百科总共有1300w+，我们这边有1000w+，由于数据不是我个人单独抓取的，无法共享，实在抱歉。可尝试多机并发抓取。

[支持\(0\)](#) [反对\(0\)](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#)网站首页。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【促销】腾讯云技术升级10大核心产品年终让利

【推荐】高性能云服务器2折起，0.73元/日节省80%运维成本

【新闻】H3 BPM体验平台全面上线



最新IT新闻:

- [郭台铭详解鸿海工业互联网战略 拟分拆在上海上市](#)
 - [扎克伯格休假照片曝光 配娃娃吃喝玩乐](#)
 - [金刚狼死侍回归漫威，迪士尼收购福克斯让好莱坞「变天」](#)
 - [乐视网发布公告：聘任刘淑青为公司总经理](#)
 - [面试软件工程师，这些准备工作你做了吗？](#)
- » [更多新闻...](#)



最新知识库文章:

- [以操作系统的角度述说线程与进程](#)
 - [软件测试转型之路](#)
 - [门内门外看招聘](#)
 - [大道至简，职场上做人做事做管理](#)
 - [关于编程，你的练习是不是有效的？](#)
- » [更多知识库文章...](#)

