##################################################################Create by: ***Davidyu***

***Last update : 2019/06/17***

***Version: V0***

##################################################################

# Version setup

## For download

In this version all the data download and save to the **database are the whole data, which means that we download all the dataset and truncate the table and store all the data,** need to update to the incremental update.

## Maven

Wget <http://mirror.bit.edu.cn/apache/maven/maven-3/3.6.0/binaries/apache-maven-3.6.0-bin.tar.gz>

tar vxf apache-maven-3.2.3-bin.tar.gz

mv apache-maven-3.6.0 /usr/local/maven3

vim /etc/profile

MAVEN\_HOME=/usr/local/maven3

export MAVEN\_HOME

export PATH=${PATH}:${MAVEN\_HOME}/bin

source /etc/profile

修改配置文件的repo地址

/usr/local/maven3/conf/settings.xml

<localRepository>/home/davidyu/software/maven/repository</localRepository>

安装常用库

mvn help:system ## run it under root and non-root user

mvn archetype:generate -DgroupId=helloworld -DartifactId= helloworld -DarchetypeArtifactId=maven-archetype-webapp -DarchetypeCatalog=internal

## Software

Spark

Hive

Mysql

Anaconda

### Chrome -headless

mkdir chrome

cd chrome

wget https://chromedriver.storage.googleapis.com/2.31/chromedriver\_linux64.zip

unzip chromedriver\_linux64.zip

cd

vi .bashrc #添加环境变量

export PATH=/home/username/chrome:$PATH #在最后一行添加后保存退出

source ~/.bashrc #立即生效

### Vundle for vim

### Tensorflow 1.12 || keras 2.2.4 general version

### Pytorch

conda create -n pytorch python=2.7

source activate pytorch

conda install pytorch torchvision -c pytorch

### Docker

[https://docs.docker.com/install/linux/docker-ce/centos/#install-docker-engine---community-1](https://docs.docker.com/install/linux/docker-ce/centos/" \l "install-docker-engine---community-1)

sudo yum install -y yum-utils \

device-mapper-persistent-data \

lvm2

sudo yum-config-manager \

--add-repo \

<https://download.docker.com/linux/centos/docker-ce.repo>

sudo yum-config-manager --enable docker-ce-nightly

sudo yum-config-manager --enable docker-ce-test

sudo yum-config-manager --disable docker-ce-nightly

sudo yum install docker-ce docker-ce-cli containerd.io

### Pycharm

修改windows hosts 文件

C:\Windows\System32\drivers\etc

|  |
| --- |
| 0.0.0.0 account.jetbrains.com |
| 0.0.0.0 www.jetbrains.com |

# Script rules

-- BASH files

## Configure file in

davidyu\_cfg.py

-save it to the python library

-create project in data\_path, analysis\_path, download\_path

## Run python file

For loop, make a python module

And run it in shell

*/home/davidyu/stock/scripts/davidyu\_stock/scripts/download/day\_history/v1*

**for i in $(seq 0 $stk\_index\_count)**

**do**

**echo $i**

**sed -r 's/TheInput/'$(echo $i)'/g' download\_data\_insert.py > to\_download.py**

**echo "python todownload"^[[>0;136;0c**

**`python to\_download.py >> $Thelog & echo $! > pidfile.txt`**

**sleep 5s**

**done**

**echo "finished" >> $Thelog**

**./run.sh > run.log 2>&1 &**

## Run log name –> to Create Running Log

Script\_name + Date + extention (py/r/scala…) + “.log

1. Run program and save the log to the current dir
2. Move the log file to the log dir

$script\_raw\_name"\_"$DATE"\_scala.log"

## Source code in shell

source ~/.bashrc

cd `dirname $0`

curr\_dir=`pwd`

programName=${0##\*/} ### filename without type e.g. test.sh > test

source $shell\_function\_dir"create\_log.sh"

## Run with log and PID

nohup ./run.sh > run.log & echo $! > pidfile.log

## Hive & Spark setup

Set mapreduce.reduce.memory.mb

Set mapred.reduce.tasks

Set io.sort.spill.percent

Set mapred.job.shuffle.input.buffer.percent

Set mapred.job.shuffle.merge.percent

Set mapred.compress.map.output

Set hive.input.format=org.apache.hadoop.hive.ql.io.CombineHiveInputFormat

Set hive.exec.dynamic.partition=true

Set hive.exec.dynamic.partition.mode=nonstrict

Set mapreduce.input.fileinputformat.split.maxsize=256000000

Set mapreduce.input.fileinputformat.split.minsize.per.node

Set mapreduce.input.fileinputformat.split.minsize.per.rack

Set hive.cli.print.header=true

Set hive.exec.compress.output

Set spark.streaming.backpressure.enabled 反压机制：如果目前系统的延迟较长，receiver端会自动减小接受数据的速率，避免系统因数据挤压过多而奔溃

### Hive 所有字符编码支持中文分区

alter database hive default character set utf8;

alter table BUCKETING\_COLS default character set utf8;

alter table CDS default character set utf8;

alter table COLUMNS\_V2 default character set utf8;

alter table DATABASE\_PARAMS default character set utf8;

alter table DBS default character set utf8;

alter table FUNCS default character set utf8;

alter table FUNC\_RU default character set utf8;

alter table GLOBAL\_PRIVS default character set utf8;

alter table PARTITIONS default character set utf8;

alter table PARTITION\_KEYS default character set utf8;

alter table PARTITION\_KEY\_VALS default character set utf8;

alter table PARTITION\_PARAMS default character set utf8;

alter table PART\_COL\_STATS default character set utf8;

alter table ROLES default character set utf8;

alter table SDS default character set utf8;

alter table SD\_PARAMS default character set utf8;

alter table SEQUENCE\_TABLE default character set utf8;

alter table SERDES default character set utf8;

alter table SERDE\_PARAMS default character set utf8;

alter table SKEWED\_COL\_NAMES default character set utf8;

alter table SKEWED\_COL\_VALUE\_LOC\_MAP default character set utf8;

alter table SKEWED\_STRING\_LIST default character set utf8;

alter table SKEWED\_STRING\_LIST\_VALUES default character set utf8;

alter table SKEWED\_VALUES default character set utf8;

alter table SORT\_COLS default character set utf8;

alter table TABLE\_PARAMS default character set utf8;

alter table TAB\_COL\_STATS default character set utf8;

alter table TBLS default character set utf8;

alter table VERSION default character set utf8;

alter table BUCKETING\_COLS convert to character set utf8;

alter table CDS convert to character set utf8;

alter table COLUMNS\_V2 convert to character set utf8;

alter table DATABASE\_PARAMS convert to character set utf8;

alter table DBS convert to character set utf8;

alter table FUNCS convert to character set utf8;

alter table FUNC\_RU convert to character set utf8;

alter table GLOBAL\_PRIVS convert to character set utf8;

alter table PARTITIONS convert to character set utf8;

alter table PARTITION\_KEYS convert to character set utf8;

alter table PARTITION\_KEY\_VALS convert to character set utf8;

alter table PARTITION\_PARAMS convert to character set utf8;

alter table PART\_COL\_STATS convert to character set utf8;

alter table ROLES convert to character set utf8;

alter table SDS convert to character set utf8;

alter table SD\_PARAMS convert to character set utf8;

alter table SEQUENCE\_TABLE convert to character set utf8;

alter table SERDES convert to character set utf8;

alter table SERDE\_PARAMS convert to character set utf8;

alter table SKEWED\_COL\_NAMES convert to character set utf8;

alter table SKEWED\_COL\_VALUE\_LOC\_MAP convert to character set utf8;

alter table SKEWED\_STRING\_LIST convert to character set utf8;

alter table SKEWED\_STRING\_LIST\_VALUES convert to character set utf8;

alter table SKEWED\_VALUES convert to character set utf8;

alter table SORT\_COLS convert to character set utf8;

alter table TABLE\_PARAMS convert to character set utf8;

alter table TAB\_COL\_STATS convert to character set utf8;

alter table TBLS convert to character set utf8;

alter table VERSION convert to character set utf8;

alter table PART\_COL\_STATS convert to character set utf8;

SET character\_set\_client = utf8 ;

SET character\_set\_connection = utf8 ;

alter table PART\_COL\_STATS convert to character set utf8;

SET character\_set\_database = utf8 ;

SET character\_set\_results = utf8 ;

SET character\_set\_server = utf8 ;

SET collation\_connection = utf8 ;

SET collation\_database = utf8 ;

SET collation\_server = utf8 ;

SET NAMES ‘utf8’;

SET NAMES 'utf8';

。

原文链接：https://blog.csdn.net/line\_aijava/article/details/79254598

## SQL script title example

-- ======================================================

-- script name : stock\_change\_interval.sql

--

-- Souce Table : stock\_dev.day\_history\_insert

--

-- Target Table : stock\_test.stock\_change\_interval

--

-- Description: Calculate the stock price change over a period P(last)-P(first)

--

-- ---------------- change log -------------------------

-- 2019-07-10 davidyu initial version

# Script framework

## Download

1. **GET url**
2. **Parse url to Dataframe**
3. **Clean the data frame**
4. **Save the data**

### Source functions

***DIR : scripts/functions***

***-ForDownload.py***

#### generate\_stock\_index.py

'''

input : i stock index raw input '601398'/ '917'

output : 601398.ss

000917.sz

'''

#### stk\_index\_list\_gen.py

'''

functions : get the stock index list without "3XXXXX" chuangyeban

output: list

list of all stock index

'''

#### CxExtractor.py

Extract contents from a html

#### combine\_csv\_in\_folder.py

#### to\_hive.scala

save the dataframe to hive

### day\_history

1. read stk index list
2. loop to download to data dir i.e. 601398.csv
3. get all csv to “all.csv”
4. put into hive

## Analysis

Do not do filter in the (linear) functions, do it before group by data

### Historical day data

#### Linear regression

/scripts/functions

**Class LinearReg**

**-def single\_linear\_reg.py one column get the slope**

#### get\_technical\_indicators

analysis/day\_history/test\_AI

--MACD

--Bollinger Bands

#### LSTM

/home/davidyu/stock/scripts/davidyu\_stock/scripts/lab/lstm

### Stock price change over a period

/home/davidyu/stock/scripts/davidyu\_stock/scripts/analysis/day\_history/increase\_value

stock\_change\_interval.sql

### networkx with the stock owner

need to follow up in the lab folder

### Machine Learning test

**Lab1:** SVM of the future N days trends (Y), X (historical price,volume, and moving averages and their numerical combination)

1,load raw history price data: ‘stock\_index, stock\_date,low,high .... adj\_close’

2.features of X and join with raw data on (stock\_date,stock\_index)

3.clean & scale the features , Y -> labels (0,1)

4.split train & test data

5.model it

6.justified the model

### All news

#### Content summarize 🡪 TextRank

### Finical report

### Owner\_liutong

Make the ratio and datetime list that some owner have

e.g. 002410 广联达 香港中央结算有限公司 6.992,6.116,2.9,2.969,2.714 2019-06-30,2019-03-31,2018-12-31,2018-09-30,2018-06-30

# Important Scripts -- for system

## Start jupyter

# 如果你的系统是装在服务器上，没有图形界面，需要远程连接的则可以加上如下参数

jupyter notebook --ip 0.0.0.0 --no-browser

## Check if chinese is in the string

zhmodel = re.compile(u'[\u4e00-\u9fa5]')

## Python

### @Property 装饰器 可读写属性 / 只读属性

### 未完成--多重继承，装饰器

#### \_\_slots\_\_

只对梳理添加某种属性

### CNTK

Keras 🡪 base source code ，need to check

<https://github.com/microsoft/CNTK>

cntk notebooks.azure.com

### Some point

Np.where(condition,x,y) if satisfied the condition, return x else y

### Bp network

<https://blog.csdn.net/weixin_40432828/article/details/82192709>

#### matrix transpose

-need no check numpy.transpose

### attr import 面向对象

### latin1 to normal

def tr(x):

x1 = x.encode('latin1',"ignore").decode('gb2312',"ignore")

return x1

# Outside data

## -新华字典全量数据

<https://github.com/pwxcoo/chinese-xinhua.git>

DIR '/home/davidyu/gits/chinese-xinhua/data'

## -中国大陆 31 个省份 1978 年至 2019 年一千多万工商企业注册信息，包含企业名称、注册地址、统一社会信用代码、地区、注册日期、经营范围、法人代表、注册资金、企业类型。

-<https://github.com/imhuster/Enterprise-Registration-Data-of-Chinese-Mainland/blob/master/Enterprise-Registration-Data/csv/README_CSV.md>

- /home/davidyu/stock/outside\_data/Enterprise-Registration-Data-of-Chinese-Mainland/csv

# Important Data

## 中国外汇&黄金储备

<http://www.safe.gov.cn/safe/whcb/index.html>

# Information

## Keras

<https://blog.csdn.net/u011746554/article/details/74393922>

Keras 网络配置

## spark-sql执行流程分析

<https://www.cnblogs.com/ulysses-you/p/9762133.html>



Keras 网络结构



Keras 预处理能力



## Parse Spark-sql script

**antlr**

# Gits & Algorithm

## 情感分析

https://github.com/SeanLee97/xmnlp.git

## the algorithms

<https://github.com/TheAlgorithms>

## 面试笔记

<https://github.com/imhuay/Algorithm_Interview_Notes-Chinese.git>

## AiLearning 🡪 AI 学习路线

<https://github.com/apachecn/AiLearning>

## Hive udf

https://github.com/klout/brickhouse.git

## \*\*numpy-ml

<https://github.com/ddbourgin/numpy-ml.git>

## LSTM

https://github.com/jaungiers/LSTM-Neural-Network-for-Time-Series-Prediction.git

# News

安邦

## 氢能

2019/06/15 日本燃料电池车和家用燃料电池发展至今，一再燃料电池和氢能技术开发灵与走在了始接前列，已卡法了日本市场应用最广泛，世界范围内技术最领先的产品。日本是一个能源极度匮乏的国家不得不重视氢能源的开发，以实现真正的能源安全和能源独立。日本政府在 能源基本计划 中将氢能源定位位与电力和热能并列的核心二次能源，病提出建设 氢能社会 的远景，希望通过氢燃料电池实现氢能在家庭 工业 交通 等灵与的应用，构建氢能产业的应用体系和消费体系。

## --2019年5月 工业产出

2019/06/18 今年5月，中国的工业产出增长降至逾17年的最低水平，一位这始接第二大经济体正在奋力应对经济下行压力。国家统计局6月15日发布的数据表明，今年5月份，中国规模以上工业增加值同比增长5%，这是自2002年2月以来的最低水平，低于前一个月的5.4%。5月的数据还显著低于路透社钓场的经济学家所预测的5.5%。

## 房地产

2019/06/19 央企大量撤出房地产行业显示资金逃离的迹象。近期，房地产行业产权转让项目明显增多，而且转让方多为氧气。根据北京产权交易所公开信息梳理统计，5月以来，共有20个房地产项目进行股权转让，而今年前4个月只有2个，去年4月至12月，9个月的时间里总共仅有10个

## 铜价秩序低迷

2019/07/10 身为制造业重要原料的铜金属近期价格持续低迷，即使在中美贸易战他暂时停火后仍不见反转，显示市场对全球经济的担忧。数据显示，铜金属价格上周帝阮福达1.7%，从4月的高点至今则下跌了11%的价格，过去十周李有巴州价格都是在下跌的。其中，中国的铜需求占全球几乎一版，而在近期中国制造业数据显现放缓态势后，市场纷纷开始怀疑刺激政策是否能维持下半年的增长，从而不看好铜金属

## 国内独立医学实验室增长迅猛，相关上市公司业绩或迎来爆发期

# PA

Query-> 去特殊字符 – 纠错- 分词- 代词消解- 词性标注- 实体识别-主谓宾提取 – 同义词替换 –专名归一化 – 预处理结果

# Papers

## LSTM

### long short term memory hochreiter 1997

### time series prediction using lstm deep neural networks

<https://www.altumintelligence.com/articles/a/Time-Series-Prediction-Using-LSTM-Deep-Neural-Networks>

## AutoML

## **NLP**

### ELMo

<https://blog.csdn.net/triplemeng/article/details/82380202>

# Book

## The links

/home/davidyu/gits/book

## [古腾堡项目](https://www.gutenberg.org/)

<https://www.gutenberg.org/>

## Data Science at the Command Line

<https://www.datascienceatthecommandline.com/preface.html>

# Method

## LightGBM

## XGBOOST

## SGD

## Loss function

<https://www.cnblogs.com/guoyaohua/p/9217206.html>

# People

## 黄学东 微软-语音识别

## 刘知远 清华-NLP

## 桥水基金、 全天候策略

## Geoffrey E. Hinton

Deep learning

# Record

## 大湾区指数基金

512973 512970