Scrapy爬虫教育部新闻标题并统计

本项目为爬取教育部新闻，使用scrapy与django结合来爬取数据

django配置

数据库

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'jiaoyu',
        'USER': 'root',
        'PASSWORD': 'Lm@599407483',
        'HOST': '127.0.0.1',
        'PORT': '3306'
    }
}
```

数据模型

对应数据库

```
class GetData(models.Model):
    title = models.CharField(verbose_name='标题', max_length=128)
    content = models.TextField(verbose_name='内容')
    add_time = models.CharField(verbose_name='添加时间', max_length=32)
    editor = models.CharField(verbose_name='编辑', max_length=32)

    class Meta:
        verbose_name = "爬虫数据表"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.title
```

以上就是django的主要配置

scrapy配置

与django连接

```
import os
import random
import sys
import django

sys.path.append(os.path.dirname(os.path.abspath('.')))
os.environ['DJANGO_SETTINGS_MODULE'] = 'jiaoyubuWenzhang.settings'
django.setup()
```

设置随机请求与请求头

```
FEED_EXPORT_ENCODING = 'UTF8'

USER_AGENT_LIST = [
    'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.139 Safari/537.36'
    "Mozilla/5.0 (Windows NT 6.2) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1062.0 Safari/536.3",
    "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1062.0 Safari/536.3",
    "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; 360SE)",
    "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.1 Safari/536.3",
    "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.1 Safari/536.3",
    "Mozilla/5.0 (Windows NT 6.2) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.0 Safari/536.3",
    "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/535.24 (KHTML, like Gecko) Chrome/19.0.1055.1 Safari/535.24",
    "Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/535.24 (KHTML, like Gecko) Chrome/19.0.1055.1 Safari/535.24"
]
USER_AGENT = random.choice(USER_AGENT_LIST)
```

爬取脚本

```
from getdata.items import GetdataItem
from scrapy.spiders import CrawlSpider, Rule
from scrapy.linkextractors import LinkExtractor


class MyJiao(CrawlSpider):
    name = "jiaoyu"
    allowed_domains = ['www.moe.gov.cn']
    start_urls = [
        'http://www.moe.gov.cn/jyb_xwfb/gzdt_gzdt/'
    ]
    #添加路由池
    for i in range(1, 10):
        url = "http://www.moe.gov.cn/jyb_xwfb/gzdt_gzdt/index_" + str(i) + ".html"
        start_urls.append(url)
    #爬取规则
    rules = [
        Rule(LinkExtractor(allow='/moe'), callback='parse_item'),
        Rule(LinkExtractor(allow='jyb_xwfb/gzdt_gzdt/s'), callback='parse_item')
    ]
    # 获取数据
    def parse_item(self, response):
        title = response.xpath('//*[@id="moe-detail-box"]/h1/text()').extract()
        add_time = response.xpath('//*[@id="moe-detail-box"]/div[1]/text()').extract()
        content = response.xpath('//*[@id="moe-detail-box"]/div[2]/p').extract()
        editor = response.xpath('//*[@id="detail-editor"]/text()').extract()

        item = GetdataItem()
        item['title'] = title
        item['add_time'] = add_time
        item['content'] = content
        item['editor'] = editor

        yield item
```

数据清洗与保存

首先设置中间件

```
from scrapy_djangoitem import DjangoItem
from app.models import GetData

class GetdataItem(DjangoItem):
    # define the fields for your item here like:
    # name = scrapy.Field()
    django_model = GetData
```

数据清洗保存

```
from app.models import GetData


class GetdataPipeline:
    def process_item(self, item, spider):
        try:
            get = GetData.objects.filter(title=item['title'][0])
            if get:
                print("数据重复")
                pass
            else:
                # print(item['add_time'][0][0:10])
                GetData.objects.create(title=item['title'][0], content=item['content'][0], add_time=item['add_time'][0][0:10],
                                       editor=item['editor'][0])
                print("save")
        except:
            print("中间件错误")
        return item
```

数据统计

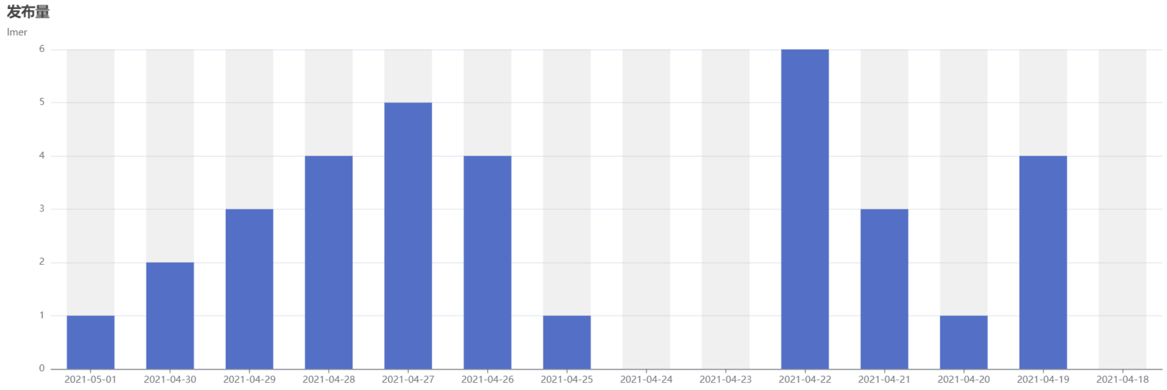
统计14天发布量

```
def day_get(d):
    for i in range(1,15):
        oneday = datetime.timedelta(days=i)
        day = d - oneday
        date_to = datetime.datetime(day.year,day.month,day.day)

        yield str(date_to)[0:10]

def Get(request):
    d = datetime.datetime.now()
    date = day_get(d)
    data = {}
    for i in date:
        data[i] = len(GetData.objects.filter(add_time=i))

    return HttpResponse(json.dumps(data))
```

前端访问接口获取数据

```
var dom = document.getElementById("container");
var myChart = echarts.init(dom);
myChart.setOption({
    title: {
        text: '发布量',
        subtext: 'lmer'
    },
    xAxis: {
        type: 'category',
        data: []
    },
    yAxis: {
        type: 'value'
    },
    series: [{
        data: [],
        type: 'bar',
        showBackground: true,
        backgroundStyle: {
            color: 'rgba(180, 180, 180, 0.2)'
        }
    }]
});
$.ajax({
    url: "/getdata/",
    type: "GET",
    success: function (res) {
        var data = JSON.parse(res);
        var time = [];
        var db = [];
        for (var key in data) {
            console.log(key);
            time.push(key);
            console.log(data[key]);
            db.push(data[key])
            myChart.setOption({
                tooltip: {
                    trigger: 'axis',
                    axisPointer: {            // 坐标轴指示器，坐标轴触发有效
                        type: 'shadow'        // 默认为直线，可选为：'line' | 'shadow'
                    }
                },
                grid: {
                    left: '3%',
                    right: '4%',
                    bottom: '3%',
                    containLabel: true
                },
                xAxis: [
                    {
                        type: 'category',
                        data: time,
                        axisTick: {
                            alignWithLabel: true
                        }
                    }
                ],
                yAxis: [
                    {
                        type: 'value'
                    }
                ],
                series: [
                    {
                        name: '当天发布量',
                        type: 'bar',
                        barWidth: '60%',
                        data: db
                    }
                ]
            });
        }
    }
})
```

效果


发布量
lmer