**海航货运物品分类模型**

**2018年1月**

**目 录**

[1. 背景 1](#_Toc505699728)

[2. 思路 1](#_Toc505699729)

[2.1 业务思路 1](#_Toc505699730)

[2.2 算法思路 1](#_Toc505699731)

[3. 实现 2](#_Toc505699732)

[3.1 百度百科爬虫 2](#_Toc505699733)

[3.2 涉及的算法 3](#_Toc505699734)

[4. 处理流程 3](#_Toc505699735)

# 背景

海航货运的数据中有上百万的运单信息，在运单中包含所运送物品的信息。但是因为运送物品都是用户填写的自由文本，无法按照物品分类来统计，去年一共运送了多少比例的电子产品，多少服装！！所以需要通过算法来做出一个相对准确的分类结果。

# 思路

## 业务思路

* 问题1：如何确定统一的分类标准？
* 解决办法：我们分别抓取了淘宝、京东、阿里巴巴以及一个百度文库上的物品分类，经过人为筛选，确定了分类标准，一共得到28个大分类,400+小分类。
* 问题2: 如何将我们的货物名对应到正确分类上去？
* 解决办法：
  + 首先使用货物名和分类名,通过互联网(百度百科)，获得更多与之相关的信息(单单的货物名提供的信息太少了)。
  + 然后我们可以计算使用货物名和分类名分别拿到的百度百科的相似性，遍历我们已有的所有分类，然后取相似度最高的那个一分类作为货物名的分类。

## 算法思路

* 问题1：如何计算两篇百度百科的相似度？
* 解决办法：将两篇百度百科变成同维度的向量，用余弦相似度来计算向量的相似度，得到的相似度就是两篇百度文档的相似度。
* 问题2：如何将百度百科变成向量？
* 解决办法：
  + 对所有要计算相似度的百度百科(很多篇)先做分词，每个词就是一个维度。
  + 对一篇百度百科做分词，使用TF-IDF算出这篇百度百科所有词的重要程度，然后得到的某一个词的重要程度就是这篇百科在这个维度上的值,然后我们就得到了一个向量。
* 问题3：如何更好地分词？
* 解决办法：将百度百科的词条作为词典，来进行分词，这样可以更好地找出一般词典里没有的词，词典需要去除英语（在词典中加入英语会导致完整的英语单词被切分）。

# 实现

## 百度百科爬虫

1. 如何爬取百度百科

百度百科有一个可以通过连续id访问的url: https://baike.baidu.com/view/{id}

需要通过不断地改变id，爬取到所有的百度百科页面。

爬取的时候，需要记录下每一个百科词条的id，url，以及文件存储路径。

1. 爬取的百科结果

1400w+的词条。

1. 解析百科
2. 在计算相似度的时候，为了扩大一个百度百科的数据量（主要是扩大分类的数据量），我们需要将与这个百度百科相关的百度百科的文本也加入到这个百度百科中。所以就需要解析百科中其他百科的链接。

方法就是直接在html里找到以百度百科（https://baike.baidu.com）开头或/开头（相对路径）的url，然后找到这些url对应的百度百科。

1. 百科中有一些关键的属性可以给之后做算法优化提供可能性，比如词条标签、分类、中文名、外文名。这些属性都需要提取出来，供以后使用。
2. 爬取下来的百度百科是html，我们需要将html解析得到对应的有用文本。

## 涉及的算法

**TF-IDF(term frequency-inverse document frequency)**

TF-IDF是一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。

**余弦相似度**

通过计算两个向量的夹角余弦值来评估他们的相似度。

# 处理流程

1. 建立起自己想要的分类：
   1. [示例](category.yml)：

|  |
| --- |
| - category\_name: 材料  sub\_category:  - 矿石  - 金属矿石  - 泥土  - 水泥  - 建设材料  - 砖  - 玻璃  - category\_name: 有色金属  sub\_category:  - 有色金属  - 合金  - 钢铁  - 五金  - category\_name: 木材  sub\_category:  - 原木  - 木柴  - 木材  - category\_name: 食品、烟酒、调味品  sub\_category:  - 食品  - 粮食  - 水果  - 酒  - 饮料  - 调味品  - 蔬菜  - 烟  - 面包  - 熟食  - 肉 |

* 1. 解释：category\_name是最后的分类名，sub\_category是小分类。之后需要对所有的小分类都做分类，然后映射到分类名上去。

1. 使用所有的分类，建立起向量（找到分类的百度百科，使用TF-IDF计算出向量）。
   1. 解释：由于向量有6w多维度，不好展示。算法详情参考[算法思路](#_算法思路)。
2. 对已有的3.3M行货物名进行分词（需要用到百度百科的条目名作为词典，详细算法参考[算法思路](#_算法思路)），得到55k个词。
   1. 图例：
      1. [原始数据](goods.csv)：

|  |
| --- |
| "测试鲜花"  "熊猫；斑马"  "SCARF"  "布卷"  "CONSOL SHIPMENT"  "GARMENTS"  "配件"  "CONSOLSHIPMENT BANDAGE"  "模具"  "普货"  "山药多酚"  "配件"  "清洁用品"  "灯"  "CONSOL"  "手袋"  "不锈钢餐具"  "GARMENTS"  "平板电脑保护套"  "M快件"  "L-CITRULLINE DL-MALATE"  "服装"  "CONSOLSHIPMENT"  "航空器接插件"  "展品"  "CONSOL SHPT SCREW"  "小刀"  "服装"  "服装"  "熊猫；斑马"  "钻"  "户外灯"  "CONSOL"  "布" |

* + 1. [处理后数据](full_cut_goods.csv)：

|  |
| --- |
| all,4945318  快件,414753  普货,196123  服装,177913  衣服,127260  拼货,120298  鲜花,80039  CONSOL,79317  配件,79009  文件,75905  电路板,61960  清单,60766  附,58302  线路板,54043  布料,48274  海鲜,44543  鞋子,43050  上衣,42065  航材,41622  鱼,37839  狗,37748  资料,34516  轴承,29504  熟食,25741  顺丰,24931  芒果,24768  混,24635  装货,24560  药品,24236  鞋,23037  鸡苗,22647  合同,22293  图纸,22253  观赏鱼,21443  蟹,21188 |

* 1. 解释：处理后的数据左边是分词名，右边是词频，我们能发现有些词出现的很频繁，这些词会是以后做分类的“重点照顾对象”

1. 对55k个词做筛选，选出词频大于3的20k个词，计算这些词和所有分类的相似度(做缓存用)
   1. 示例：

|  |
| --- |
| 快件 0.040757883 0.033496283 0.008741993....  普货 0.015684761 0.015941916 0.006197242....  服装 0.021555627 0.026448764 0.01318887....  衣服 0.016234517 0.027101524 0.01927977....  拼货 0.084111825 0.120638266 0.028365549....  鲜花 0.04665393 0.07389851 0.023906484....  CONSOL 0.18211387 0.33964866 0.01717055....  配件 0.075554945 0.0905541 0.02065059....  文件 0.023815818 0.04508799 0.003791385....  电路板 0.15431656 0.1888506 0.066382185....  清单 0.13523293 0.08828531 0.013413044....  附 0.03172076 0.047363315 0.0064427624....  线路板 0.15431656 0.1888506 0.066382185....  布料 0.031033326 0.045353487 0.025554623....  海鲜 0.02095027 0.029393291 0.014795702....  鞋子 0.005877837 0.012039162 0.018988255....  上衣 0.04758468 0.044817068 0.02236361....  航材 0.0037374373 0.0119285965 0.01128729....  鱼 0.01976766 0.021212509 0.00960493....  狗 0.0089824395 0.010902214 0.0030800656....  资料 0.044838373 0.030943362 0.0376428....  轴承 0.069537126 0.12853193 0.07770968....  熟食 0.045629628 0.054112237 0.029868232....  顺丰 0.022166118 0.025011964 0.009115014....  芒果 0.031247921 0.041175194 0.01666762....  混 0.037363276 0.025080238 0.011819851....  装货 0.010500036 0.022184411 0.010257284.... |

* 1. 解释：每个词都会有一组相似度，和我们一开始选定好的分类一一对应。

1. 对3.3M行货物进行去重，得到155k行货物名，分别对每一行进行分词。
   1. [示例](row_cut_goods.csv)：

|  |
| --- |
| "测试,鲜花",测试鲜花  "熊猫,斑马",熊猫；斑马  SCARF,SCARF  布卷,布卷  "CONSOL,SHIPMENT",CONSOL SHIPMENT  GARMENTS,GARMENTS  配件,配件  "CONSOLSHIPMENT,BANDAGE",CONSOLSHIPMENT BANDAGE  模具,模具  普货,普货  "山药,多酚",山药多酚  清洁用品,清洁用品  灯,灯  CONSOL,CONSOL  手袋,手袋  不锈钢餐具,不锈钢餐具  "平板电脑,保护套",平板电脑保护套  "邮件,快件",M快件  "活体动物,-,CITRULLINE,DL,-,MALATE",L-CITRULLINE DL-MALATE  服装,服装  CONSOLSHIPMENT,CONSOLSHIPMENT  "航空器,接插件",航空器接插件 |

* 1. 解释：图中每一行会有两列，左边是切词后的结果，右边是原始数据，保留原始数据的目的是为了之后做映射的时候，能直接把原始数据映射为分类。

1. 将一行中的几个词分别做分类的相似度计算，然后将得到的相似度取平均值,最后找到相似度最大分类（取平均值只是一种策略，也可以取最大值，或者直接给一行货物名多个分类）
   1. 示例：
      1. [可读](result.txt)：

|  |
| --- |
| CONSOLIDATION ALUMINUM PART SILICON PART ---- 海鲜  贴片箱包扣 ---- 纺织品、衣被鞋帽  牛仔裤男上衣 ---- 纺织品、衣被鞋帽  测试鲜花 ---- 植物  压力泵轴承轴 ---- 工业机械、电器、仪器、数码产品  豌豆提取物 ---- 植物  COTTON LACE ---- 纺织品、衣被鞋帽  CONSOL LASER MFP LASER PRINTER ---- 工业机械、电器、仪器、数码产品  汽车前杠 ---- 交通工具  CONSOL FISHING LINE ---- 工业机械、电器、仪器、数码产品  设备、册子、标书 ---- 工业机械、电器、仪器、数码产品  PAPERBAG ---- 工业机械、电器、仪器、数码产品  CONSOLIDATED AS PER ATTACHED MANIFEST COAT JACKET BRA THONG ---- 工业机械、电器、仪器、数码产品  SAMPLE OF CHILD SATETY SEAT ---- 纺织品、衣被鞋帽  CONSOL AS PER ATTACHED MANIFEST LIGHTING FITTINGS ---- 混装物  碟形弹簧 ---- 交通工具  SMOKE DETECTOR HEAT DETECTOR ---- 海鲜  喷油头 ---- 工业机械、电器、仪器、数码产品  CONSOL ALUMINIUM ALLOY DIE CASTING PARTS ---- 有色金属  熊猫；斑马 ---- 植物  SCARF ---- 纺织品、衣被鞋帽  布卷 ---- 工业机械、电器、仪器、数码产品  CONSOL SHIPMENT ---- 混装物  GARMENTS ---- 纺织品、衣被鞋帽 |

* + 1. [分析](verbose.txt):

|  |
| --- |
| CONSOLIDATION ALUMINUM PART SILICON PART --- ['CONSOLIDATION', 'ALUMINUM', 'PART', 'SILICON', 'PART'] --- ['name:海鲜 sub:虾类 sim:0.3667121', 'name:有色金属 sub:合金 sim:0.34807974', 'name:工业机械、电器、仪器、数码产品 sub:工业机械 sim:0.28134432', 'name:材料 sub:建设材料 sim:0.18779996', 'name:植物 sub:植物 sim:0.18134871']  贴片箱包扣 --- ['贴片', '箱包', '扣'] --- ['name:纺织品、衣被鞋帽 sub:服装 sim:0.27386013', 'name:纺织品、衣被鞋帽 sub:衣服 sim:0.24675946', 'name:纺织品、衣被鞋帽 sub:纺织品 sim:0.24094124', 'name:工业机械、电器、仪器、数码产品 sub:仪器 sim:0.2246237', 'name:工业机械、电器、仪器、数码产品 sub:家电 sim:0.22431256']  牛仔裤男上衣 --- ['牛仔裤', '男', '上衣'] --- ['name:纺织品、衣被鞋帽 sub:服装 sim:0.56927836', 'name:纺织品、衣被鞋帽 sub:衣服 sim:0.4618534', 'name:纺织品、衣被鞋帽 sub:鞋 sim:0.25747454', 'name:纺织品、衣被鞋帽 sub:纺织品 sim:0.19396228', 'name:食品、烟酒、调味品 sub:肉 sim:0.19102125']  测试鲜花 --- ['测试', '鲜花'] --- ['name:植物 sub:植物 sim:0.46242264', 'name:工业机械、电器、仪器、数码产品 sub:仪器 sim:0.3361314', 'name:植物 sub:灌木 sim:0.22001716', 'name:工业机械、电器、仪器、数码产品 sub:电器 sim:0.21391493', 'name:材料 sub:建设材料 sim:0.18016604']  压力泵轴承轴 --- ['压力泵', '轴承', '轴'] --- ['name:工业机械、电器、仪器、数码产品 sub:电动机 sim:0.43582153', 'name:工业机械、电器、仪器、数码产品 sub:内燃机 sim:0.35689005', 'name:有色金属 sub:钢铁 sim:0.2794145', 'name:有色金属 sub:合金 sim:0.2745563', 'name:工业机械、电器、仪器、数码产品 sub:仪器 sim:0.27218804']  豌豆提取物 --- ['豌豆', '提取物'] --- ['name:植物 sub:植物 sim:0.7401058', 'name:食品、烟酒、调味品 sub:蔬菜 sim:0.35642523', 'name:医药品 sub:中成药 sim:0.21920402', 'name:医药品 sub:中药材 sim:0.21753564', 'name:食品、烟酒、调味品 sub:水果 sim:0.1885316']  COTTON LACE --- ['COTTON', 'LACE'] --- ['name:纺织品、衣被鞋帽 sub:纺织品 sim:0.45588106', 'name:纺织品、衣被鞋帽 sub:服装 sim:0.40692744', 'name:纺织品、衣被鞋帽 sub:床上用品 sim:0.3402771', 'name:工业机械、电器、仪器、数码产品 sub:工业机械 sim:0.30028018', 'name:海鲜 sub:虾类 sim:0.25381148']  CONSOL LASER MFP LASER PRINTER --- ['CONSOL', 'LASER', 'MFP', 'LASER', 'PRINTER'] --- ['name:工业机械、电器、仪器、数码产品 sub:电子计算机 sim:0.27953154', 'name:混装物 sub:CONSOL sim:0.2342434', 'name:工业机械、电器、仪器、数码产品 sub:数码 sim:0.22762764', 'name:工业机械、电器、仪器、数码产品 sub:家电 sim:0.22176766', 'name:工业机械、电器、仪器、数码产品 sub:仪器 sim:0.18996376'] |

* 1. 解释：这一步需要产生两个结果
     1. 一个是人能很容易读取的数据，方便之后人为地判别分类是否准确。
     2. 一个是具体的分类情况，也就是需要包含做分类时，各个的相似度，这样可以方便我们分析，进行算法的提升。
  2. 结果：看了50条的数据，可以发现现在的分类结果有
     1. 55%准确率（算上英语的条目）
     2. 62%准确率（不算上英语条目）