**实体识别训练数据的BIO自动标注**

## 前言：

NLP (Natural Langunge Possns,自然语言处理)是计算机科学领域以及人工智能领域的一个重要的研究方向，它研究用计算机来处理、理解以及运用人类语言(如中文、英文等)，达到人与计算机之间进行有效通讯。随着近几年人工智能的热潮，各个行业对于NLP（自然语言处理）技术的需求越来越强烈，通过NLP技术来进行命名实体的识别是一个非常基础的业务需求，尤其在专业领域的需求非常强烈；本文作者因为需要针对于汽车领域的数据进行实体识别，所以以下的所有的内容都是围绕着汽车类数据进行说明，其它领域的数据处理方式基本相同；

随着机器学习的热潮的兴起，很多领域都开始尝试通过AI来进行行业的创新和应用，AI虽然可以处理很多人类不能够完成的事情，但前题是需要大量的基础数据进行模型的训练工作，而对于数据格式化标注就是AI的开始，目前很多行业都留存有大量的数据，但如何把这些历史数据快速的标记完成供机器学习进行使用，是遇到的最大的难题，目前通行的办法是通过人工进行标注，但人工标注有几个问题是无法解决的；一个是人力成本的问题，海量数据的标注势必需要大量的人工，而在今天人力成本变得越来越昂贵，所以成本的支出使很多人望而却步。其次是效率问题，人工标注的效率非常低尤其在处理文本内容时，一个熟练的人其每小时的BIO标注内容也仅为2000-3000p字，一天能够处理的量也就是万字左右；每三就是错误率问题，机器学习的准确与否与训练素材的质量有直接的相关性，而人工标注的内容缺少必要的复核校对，所以其准确度无法进行评测，所以其会对于最终的结果产生影响；

为了解决目前在标注中存在的问题，根据行业的规律提取出行业的字库，然后通过不断的优化字库并通过算法来自动的生成标注信息变得尤其重要；

本发明就是作者通过建立汽车行业的实体标识库，然后通过相关的多种算法来完成原始数据自动标注过程。

## 系统的架构及说明

1. 系统架构：



1. 架构说明：

系统的架构比较简单，共有4个处理环节：

Step-1：对于原始素材的格式化处理，因为原始素材由很多碎小文件组成，需要手工把这些小文件拼成一个大文件，另外还需要对于文件中的特殊字符进行处理，使文件能够正常的被处理；

Step-2：多种算法进行信息的标注，首先对于手机号码信息进行分析，其实还需要对于其中的金额信息进行分析，最重要的是根据实体库中的数千实体命名信息进行分析，最后生成带有实体分析信息的内容文件；

Step-3：根据Step-2中带有实体名称分析信息的文件生成BIO标注信息的格式文件；

BIO格式信息说明：

序列标注（Sequence labeling）是我们在解决NLP问题时经常遇到的基本问题之一。在序列标注中，我们想对一个序列的每一个元素标注一个标签。一般来说，一个序列指的是一个句子，而一个元素指的是句子中的一个词。比如信息提取问题可以认为是一个序列标注问题，如提取出会议时间、地点等。

序列标注一般可以分为两类：

1、原始标注（Raw labeling）:每个元素都需要被标注为一个标签。

2、联合标注（Joint segmentation and labeling）：所有分段被标注为同样标签。

命名实体识别（Named entity recognition， NER）是信息提取问题的一个子任务，需要将元素进行定位和分类，如人名、组织名、地点、时间、质量等。

举个NER和联合标注的例子。一个句子为：Yesterday , George Bush gave a speech. 其中包括一个命名实体：George Bush。我们希望将标签“人名”标注到整个短语“George Bush”中，而不是将两个词分别标注。这就是联合标注。

解决联合标注问题的最简单的方法，就是将其转化为原始标注。标准做法就是使用BIO标注。

BIO标注就是将每个元素标注为“B-X”、“I-X”或者“O”。其中，“B-X”表示此元素所在的片段属于X类型并且此元素在此片段的开头，“I-X”表示此元素所在的片段属于X类型并且此元素在此片段的中间位置，“O”表示不属于任何类型。

比如，我们将 X 表示为名词短语（Noun Phrase, NP），则BIO的三个标记为：

（1）B-NP：名词短语的开头

（2）I-NP：名词短语的中间

（3）O：不是名词短语

## 系统的主要功能

1. 手机号码及金额信息的提取：

该部分主要是通过正则表达式来完成识别的替换工作

1. 手机号码的正则表达式为:

RegExp(“1[34578][\\d{9}](file://d%7b9%7d)”)

1. 带万字的金额，如：32万:

RegExp(“\\d{2,3}万”)

1. 带格式的金额，如：13，234:

RegExp(“\\d{2,4},\\d{3}”)

1. 不带格式纯数字金额，如：1000:

RegExp(“\\d{2,4}”)

正则表达式：正则表达式是对字符串（包括普通字符（例如，a 到 z 之间的字母）和特殊字符（称为“元字符”））操作的一种逻辑公式，就是用事先定义好的一些特定字符、及这些特定字符的组合，组成一个“规则字符串”，这个“规则字符串”用来表达对字符串的一种过滤逻辑。正则表达式是一种文本模式，该模式描述在搜索文本时要匹配的一个或多个字符串。

?,\*,+,\d,\w 都是等价字符  
?等价于匹配长度{0,1}  
\*等价于匹配长度{0,}   
+等价于匹配长度{1,}  
\d等价于[0-9]

\D等价于[^0-9]  
\w等价于[A-Za-z\_0-9]

\W等价于[^A-Za-z\_0-9]。

1. 实体关键字的标识比对：

实体关键字分类为如下几个：品牌（BRAND）、车系(VEHICLE)、车型（MODLE）、关注点(FOCUS)、配件(SPARE)、付款方式（PAY）、动力(POWER)；

根据汽车之家等专业网站的展示数据，整理后形成实体关键字库，目前收集的实体关键字库有效记录为3000+个；

把实体关键字内容存入Mysql数据库中，其表结构为：ent\_name(实体关键字), tag（标识内容）。

处理过程通过全局匹配方式进行查找替换，以完成实体关键内容的标识；

其关键代码为：

try:

inf = open(infile,'r')

sf = open(swpfile, 'w')

except:

print('文件打开错误')

exit(2)

for line in inf.readlines(): #依次读取每行

if not line:

continue

line = line.strip() #去掉每行头尾空白

line = ex\_mobile(line) #提取电话号码

line = ex\_48phone(line) #处理400和800电话

line = ex\_price1(line) #处理带万的金额

line = ex\_price2(line) #处理带,的金额

line = ex\_price3(line) #处理一般的金额

line = ex\_keyword(line) #处理关键字的内容

sf.write(line+"\n") #因为BIO的需求，每行中间有一个换行

inf.close()

sf.close()

1. 生成BIO格式的文件：

根据转换后的带标识信息的文件，处理成BIO格式的标注文档，BIO格式要求每个字为一行，所以需要把每行的内容，按字进行分割，并转成多行，内容，具体的关键代码如下：

try:

sf = open(swpfile, 'r')

bf = open(bfile, 'w')

except:

print('文件打开错误')

exit(2)

for line in sf.readlines():

if not line:

continue

mlist = list(line)

if mlist[0] == '[':

line = line.replace("[", "\n")

line = line.replace("]", "\n")

line = line.replace(",", "\n")

bf.write(line)

else:

if " " in mlist:

mlist.remove(" ")

line1 = " O\n".join(mlist)+"\n"

bf.write(line1)

sf.close()

bf.close()

通过此转换后生成的内容，即可直接进行机器学习；

## 主要的成果

通过该自动实体标注程序的使用，标识信息的准确度达到100%，每1000万条数据的标注时间为30分钟，字数量达1.2亿多，人工标注的效率的指数倍增长，大大节约了成本和缩短了时间，解决了标注信息的工作量问题，这种方式也可应用于其它各个专业领域的数据信息的标注工作。