【NLP】非监督文本匹配算法——BM25 Python实现

原创 一笑清寒 皮皮AI记 2022年04月18日 07:01

文 | 菊子皮 (转载请注明出处)
B站：科皮子菊

皮皮AI记
AI算法工程师，CSDN博客专家。
186篇原创内容

公众号

算法原理与程序使用

BM25算法原理参见我的博文：【NLP】非监督文本匹配算法——BM25，代码已上传至Github：https://github.com/Htring/BM25[1]，有兴趣的可以查看源码。

测试程序：

```
bm25 = BM25()
result = bm25.cal_similarity("自然语言处理并不是一般地研究自然语言")
for line, score in result:
    print(line, score)
```

测试结果如下：

```
自然语言处理是计算机科学领域与人工智能领域中的一个重要方向。 1.012567843290477
它研究能实现人与计算机之间用自然语言进行有效通信的各种理论和方法。 2.0911221271793545
自然语言处理是一门融语言学、计算机科学、数学于一体的科学。 1.012567843290477
因此，这一领域的研究将涉及自然语言，即人们日常使用的语言， 2.2068046420905443
所以它与语言学的研究有着密切的联系，但又有重要的区别。 1.4616618736274032
自然语言处理并不是一般地研究自然语言， 3.2072055608059036
而在于研制能有效地实现自然语言通信的计算机系统， 1.201522188129132
特别是其中的软件系统。因而它是计算机科学的一部分。 0
在信息搜索中，我们做的第一步就是检索。 0
再延展一下，搜索这项功能在我们生活中也是太多太多。 0
```

```
大众一点就是搜索引擎，商品搜索等，在问题系统中可以匹配相似的问题，然后返回对应答案等。 0
文本匹配包括监督学习方法以及非监督学习方法。 0
或者分为传统方法和深度学习方法。 0
BM25 在 20 世纪 70 年代到 80 年代被提出，到目前为止已经过去二三十年了，但是这个算法依然在很多信息检索的任
有时候全称是 Okapi BM25，这里的“BM”是“最佳匹配”（Best Match）的简称。 0
那么，当通过使用不同的语素分析方法，语素权重判定方法以及语素与文档的相关性判定方法，可以衍生很多不同的搜索
```

其中第一列为待检索的内容，第二列为query与待检索内容之间的打分。

程序设计简要思路

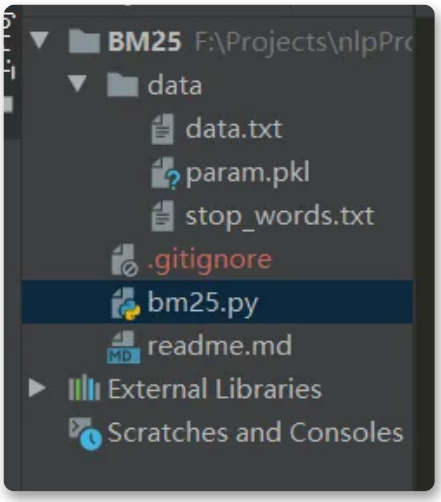
待检索的数据存放在data/data.txt中，如果BM25这个类在初始化时没有传入一个文档路径，就使用这个默认路径。

程序在第一次加载时会对待检索的数据进行统计计算，然后将相关参数保存到data/param.pkl中。当更换文档时，需要收到删除data/param.pkl。

程序中使用jieba进行分析，构建语素。使用data/stop_words.txt作为程序的停用词。

BM25中的 `cal_similarity()` 方法提供对外的计算文本相似度（文本匹配）接口，其中包含一个参数，即待搜索的query，其返回的结果是list，list中包含tuple(doc, score)形式的内容，代表各文档与query之间的打分值。

程序结构图如下：



BM25核心算法程序如下：

```
def _cal_similarity(self, words, index):
    score = 0
    for word in words:
        if word not in self.param.f[index]:
```

```
            continue
        molecular = self.param.idf[word] * self.param.f[index][word] * (self.param.k1 + 1)
        denominator = self.param.f[index][word] + self.param.k1 * (1 - self.param.b +
                                                                   self.param.b * self.param.l
                                                                   self.param.avg_length)
        score += molecular / denominator
    return score
```

Reference

[1] https://github.com/Htring/BM25: https://github.com/Htring/BM25

END

往期推荐

剑指Offer刷题集| Python基础

更多精彩文章

非监督文本匹配算法——BM25
基于Pytorch lightning与BiLSTM-CRF的NER实现
基于BiLSTM-CRF的序列标注
基于隐马尔可夫模型（HMM）的命名实体识别（NER）实现