2023年东华理工大学数学建模竞赛

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与本队以外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的, 如果引用别人的成果或其它公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们愿意承担由此引起的一切后果。

参赛题号（从A/B/C中选择一项填写）：

参赛队报名号：53

是否愿意参加暑假数学建模集训：愿意（愿意或不愿意）

参赛队员信息:

	参赛队员1	参赛队员2	参赛队员3
姓名	黄镁豪	董政	谢宗晟
学号	2021213357	2021213195	2022213124
学院	软件学院	软件学院	软件学院
专业	软件工程	软件工程	软件工程

小学数学应用题相似性度量及难度评估

摘要

为了更好的实现

目录

1 问题重述与分析

1.1 问题背景

互联网迅猛发展，线上教育平台这种新型教育模式逐渐兴起。各种基于互联网的教育模式渐渐的发展起来了。利用互联网的高度便利性和自定义性，因材施教的程度得到了进一步发展。为了进一步实现用户的个性化学习，某MOOC在线教育平台提供了个性化题库的功能。该题库系统会记录用户的学习过程，而自动生成对应的课后习题。但该系统目前来说还存在着很大缺陷。

题目系统为了实现个性化试题，主要是实现两个子功能：**相似度评估系统**和**难度评估系统**。

目前而言，这两个系统都有明显的缺陷。

1.1.1 相似度评估系统

该系统中，评判两个题目之间相似度主要依据是**题干文字**和**事先标注题目的知识点信息**。前者无法应对不同表述但是解法相同的题目，后者与知识点划分方式相关，难以达到真正的拓展练习的地步，这急需要改进。

1.1.2 难度评估系统

该系统中，判断题目难度的依据主要是**考试的类型**和**教师的主观经验**。这两种方式的局限性都太大了，前者只能够判断考试试题的难度，然而还有更多的题目是不会出现在考试试题中的；后者太主观了，不同的老师可能会给出完全不同的两种回答。并且，一个题目难度和题面的表达、学习者的状态、学习者的知识储备等等因素有关。所以，该系统仍然需要进一步改进。

1.2 问题分析

1.2.1 关于相似度和难度评估的研究现状

当今学术界对难度的评估大都集中在?

1.2.2 基于NLP理论和DBSCAN聚类的相似度检验

1.2.3

2 模型假设与符号说明

2.1 模型假设

- 假设

2.2 符号说明

3 模型的建立与求解

算法描述 1

1. 打开冰箱
2. 把大象放进去
3. 关上冰箱

3.1 相似性度量模型

3.1.1 度量角度的确立

在对小学数学应用题的相似性度量过程中，通常会使用两个依据：

1. 根据题干文字进行相似性比对，确定两道题目间题面的相似程度。
2. 通过人工或机器学习的方式为题目根据知识点进行“标签化”，两道题目之间的标签重合越多，则越相似。

实际上，进行“标签化”对题目进行相似性度量是较为科学且准确的，符合师生快速锁定同类题型进行巩固训练的实际需求，而光凭文本信息进行相似性度量的结果并不符合实际的教育需求。

但仅根据少量题目样本无法将“标签化”过程有效自动化，难以避免通过人工手段为题目加上知识点标签。考虑到平台运营的切实情况，人工进行“标签化”操作成本过大，且判断过程中人为主观因素较多，可能也会导致度量结果出现较大偏差。

因而，本文经过综合考虑，还是选择通过题干文字进行相似性比对进行相似性度量。

3.1.2 度量过程的关键步骤

本文将度量过程总结成四个关键步骤以便于读者理解，下文也将围绕这四个关键步骤进行展开：

1. 文本预处理

   文本预处理是自然语言处理中的重要步骤之一，其目的是将原始的文本数据转换成计算机可以处理的形式。为了提高文本处理任务的效率和准确性，文本预处理是在进行后续分析处理的必要前置步骤。

2. 借助LDA模型建立相似性矩阵

3. 使用余弦相似度计算相似性结果

3.1.3 文本预处理

首先需要对题目文本进行分词处理，以好通过词袋模型的方式将题目文本进行向量化，进行进一步的研究。因本文研究的题目大多处于中文语言环境下，故笔者考虑采用NLPIR分词系统对题目进行分词处理。本文为了研究方便，使用的是被广泛使用的开源中文分词工具jieba。另外也可以使用NLPIR分词系统达到相同效果。NLPIR分词系统是由中国科学技术大学自然语言处理与社会人文计算实验室开发的一款中文自然语言处理工具。它是基于统计和规则两种方法相结合的分词系统，能够对中文文本进行精准的分词和词性标注，完美契合本文的研究需要。

在分词处理的过程中，还需要分离并忽略标点符号、数字、停用词等词语的影响。详细的处理操作可以参考周萍老师在《语义分析及相似性度量方法》研究中总结的预处理流程。但本文简化了处理流程，仅对分词结果进行了简单的标点符号排除与数字排除，以加快研究进程。

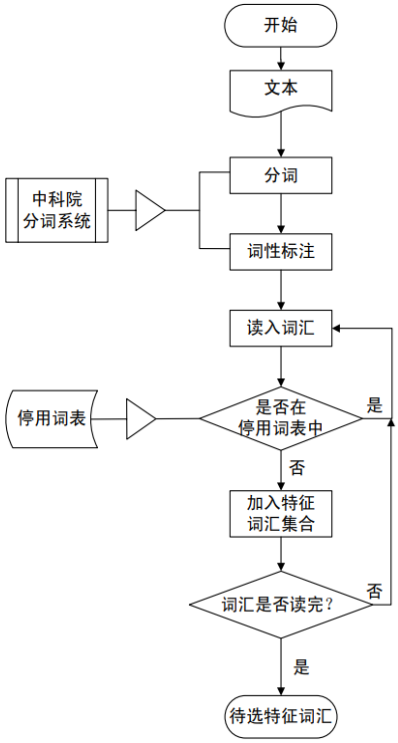

图 1: 文本预处理流程图

在这里也给出一段进行文字预处理的Python代码以作参考，并作为后续研究的前提：

代码 1　Python 文字预处理代码

```
def chinese_word_cut(mytext):
    jieba.load_userdict(dic_file)
    jieba.initialize()
    try:
        stopword_list = open(stop_file,encoding ='utf-8')
    except:
        stopword_list = []
        print("error in stop_file")
    stop_list = []
    flag_list = ['n','nz','vn']
    for line in stopword_list:
        line = re.sub(u'\n|\\r', '', line)
        stop_list.append(line)

    word_list = []
    #jieba分词
    seg_list = psg.cut(mytext)
    for seg_word in seg_list:
        word = re.sub(u'[^\u4e00-\u9fa5]','',seg_word.word)
        find = 0
        for stop_word in stop_list:
            if stop_word == word or len(word)<2:
                    find = 1
                    break
        if find == 0 and seg_word.flag in flag_list:
            word_list.append(word)
    return (" ").join(word_list)
```

3.1.4 文本预处理

代码 2　C++ Hello World!

```
    代码内容
```

3.2 基于模糊数学的难度度量模型

相对于

3.3

4 模型的评价与改进

4.1 模型的优点

- 12

4.2 模型的缺点

4.3 模型的改进

A 主要使用的软件

1. 文字编辑方案：Visual Studio Code + LaTeX+ Git + Zotero
2. 程序模拟：PyCharm + Python
3. 绘图软件：XMind + PyCharm + Python + GeoGebra

B 程序代码

代码 3 计算TF-IDF值