暑期学校实验项目：高考志愿填报助手

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 小组名称 | AI算法A组 | | | | | | |
| 姓 名 | 邵一展 | 专业 | 人工智能专业 | 班级 | 091181 | 学号 | 09118122 |
| 实验时间 | 2020.8.31-2020.9.23 | | 指导教师 | 孔祥龙 | | 成绩 |  |
| **一、实验背景和目的**  高考志愿填报助手意在帮助高考完的考生根据自己所在省份的预测自身适合的大学与专业，有助于考生更加了解自己、明确目标定位。上学期，我们收集了各个省份的一分一段表以及部分高校的专业录取分数线，短学期内，我们组的任务目标为帮助考生预测可选大学，确定其冲一冲，稳一稳，保一保的若干所大学。 | | | | | | | |
| **二、小组任务和个人任务**  小组任务：预测学校。具体实施方案为根据各个省份的一分一段表以及88所高校的专业录取分数线，随机模拟若干名考生成绩（名次）作为预训练数据，再利用神经网络预测输出，还可以对输出进行筛选等副操作。  个人任务：负责随机模拟第75-88所学校的数据，共260000+条，每条数据由名次、省份、文理类别、所选大学所组成。 | | | | | | | |
| **三、个人任务需求分析**  上学期，我们收集了88所高校的各专业录取最低分数线。我主要负责其中75-88所。  根据高校各个专业的最低分数线的收集表，按省份、类别、年份划分，确定每一省份每一年每一类别的录取专业录取最低分分中的最高最低分，再根据人工确定的附加分模拟数据。  例如东南大学在山东理科2018年时专业录取的最低分中分数最高为667分（类似于建筑等），最低为620分，同时我们为最高分增加一个附加分（例如20分），最终代表620-687（667+20）的同学都有较大机会考到东南大学，该附加分的存在也保证了预测时候的稳一稳、保一保策略，保证有高分的同学可以有保底高校选择。（因为我们只是预测学校，故忽略了专业的特殊性，只要考生高于620分理应为其推荐东南大学，不再考虑其目标专业）  但要注意的是，考虑到山东理科2018年620-687区间的同学并不全是考上东南大学的学生，于是我们对此分数区间进行了抽样，随机在区间内挑选N/20名同学假设其考上了东南大学。（N代表此区间总人数，由该省份该类别该年份的一分一段表所确定，即山东理科2018.json，如何获得json的名字详见实验过程的关键代码）  此外，我们还有人工设置模拟数据的环节，主要考虑到清华北大的特殊性以及json文件不完备等情况。将超出一分一段表的分数我们设置为nan，将不存在的一分一段表我们设置为null，在最后生成的csv中我们需要人工根据其他省份的数据对nan或者null的数值进行近似。 | | | | | | | |
| **四、实验过程（需附上关键代码及相关说明）**  college\_data = pd.read\_csv('大学信息.csv') # 读取大学数据  college\_batch\_data = college\_data.tail(15) # 取后15条数据，即负责的第75-88所大学  university = college\_batch\_data['collegeName'].values.tolist()  分别为西北大学，西安电子科技大学，长安大学，兰州大学，新疆大学，石河子大学，青岛大学，广州大学，中国矿业大学（北京），中国地质大学（北京），北京联合大学，复旦大学医学部，重庆大学，中国科学技术大学，浙江工业大学。  result = open("09118122邵一展.csv","w",encoding="utf-8-sig",newline='')  writer=csv.writer(result)  writer.writerow(["rank","province","category","college"])  确定最后csv的存放位置，为09118122邵一展.csv，并写入column名称。  for u in university:      for y in Year:          for p in Province:              for c in category:                  temp\_data = csv\_data.loc[(csv\_data["College"] == u) & (csv\_data["Province"] == p) &                                          (csv\_data["Year"] == y) & (csv\_data["category"] == c),                                          ["College", "Year" , "Province", "category","score"]]  优先确定university，Year，Province，category四个列表，并进行for循环，筛选数据并存到temp\_data中。本目的意在为后续确定最高最低分作准备。  score\_list = temp\_data["score"].values.tolist()  score\_list.sort()  low\_score = score\_list[0] #录取最差专业的最低分，如xx的录取最低分  high\_score = score\_list[-1] + addition #录取最好专业的最低分加上向上浮动的分值  确定最高最低分后，随机抽样N/20名同学。  if c == "all": temp = "json/" + str(y) + p + "all" + ".json"  else: temp = "json/" + str(y) + p + c + ".json"  该部分代表了如何从省份、年份、类别确定json名字，读取后放入load\_dict中。  if (str(low\_score) in load\_dict[str(y)][p][c]) & (str(high\_score) in load\_dict[str(y)][p][c]):  low\_rank = load\_dict[str(y)][p][c][str(low\_score)] #最低分的名次  high\_rank = load\_dict[str(y)][p][c][str(high\_score)] #最高分的名次  number = (low\_rank-high\_rank) // 20 + 1 #至少一名，近似该分数段有20/88的学生去该大学  for \_ in range(number):  random\_score = random.randint(low\_score, high\_score) #产生随机分数  if (str(random\_score) in load\_dict[str(y)][p][c]):  random\_rank = load\_dict[str(y)][p][c][str(random\_score)]  其余情况代表超出一分一段表的分数或不存在一分一段表，我们分别对两种情况设置为nan和null，并人工模拟。  （已保留主干内容。对json和csv的相关处理已忽略。） | | | | | | | |
| **五、实验结果与分析**  rank, province, category, college  1503, 北京, 文科, 西北大学  1186, 北京, 文科, 西北大学  1117, 北京, 文科, 西北大学  ……  19023, 四川, 理科, 浙江工业大学  15684, 四川, 理科, 浙江工业大学  13894, 四川, 理科, 浙江工业大学  我们数据输出为csv形式，column分别是学生的名次、学生所在省份、学生所属类别（文科/理科/不分文理）、最后所去大学。例如第一行（1503, 北京, 文科, 西北大学）即代表这名模拟出来的学生为北京文科考生（年份已忽略），全市排名为第1503名，且最后去了西北大学（专业不定）。  Csv中共267782条数据，数据的数量主要由高低分数区间段内的人数所决定，原因是我们没有途径获得每年每个高校对各个省份招收各个类别的学生的数量，故只能近似预测N/20的人会被录取。  模拟的数据经过包装整合、预处理归一化等操作后，便可以作为训练数据输入到神经网络中进行初始训练了。 | | | | | | | |
| **六、实验总结与心得体会**  本次高校志愿填报项目，我们组主要进行AI算法实现部分，意在根据各省的一分一段表和高校录取成绩模拟上百万的学生数据，实现学校的预测功能。我在其中主要负责安排分工和模拟第75-88所大学的数据。  通过本次实验，除了独立编写代码的能力提升，还增强了我的团队意识。除了小组内分工合作、均衡划分代码量、共同实现预测学校的AI算法之外，我还看到了小组间的相互协作，例如与第一组数据收集的同学对接，例如与第九组、第十组的前后端及数据库的对接等等，这些分工协作都有利于我更加了解到，一个项目是如何一步步从需求分析到系统设计，从项目准备到持续集成而逐渐完善起来的。  在本次项目中，团队协作显得尤为重要，几次课上课下的积极讨论、策略协调与分工协同都无一例外地为我们小组任务做了充分的准备，分工说明再三修改、对接bug持续修复，这些问题都逐渐push我们的任务不断前进。  还记得刚开始我们接到任务时的迷茫，完全不知道如何用AI算法进行预测，也不知道我们的输入数据会是什么样、输出数据又要是什么样，到最后任务逐渐清晰、目标逐渐明朗时，大家又都齐心协力将数据模拟了出来并进行了神经网络的训练。我们一开始提出的选择地域、选择是否985、211、双一流等等目标也都实现，一股成就感也油然而生。不过遗憾的是，因为小组过多，无法详尽地了解每个小组背后的机制原理，颇为可惜，也希望还有机会可以再次尝试其他组的工作，体验全新的氛围，学习全新的知识。 | | | | | | | |

2020年9月制