

# HW3 选择二

## 18340013 陈琮昊

### 理论部分：

**理论部分：**习题 12.1 假设一个 DTMC 对一个离散并且有  $N$  个可能取值（状态）的随机变量  $X$  的演化进行建模。证明我们需要  $N^2 - 1$  个数来完全描述这个 DTMC。

证明：

对于一个离散且有  $N$  种取值的随机变量，用马尔科夫链描述该随机过程时，首先有一个转移概率矩阵，其规模为  $N \times N$ ，由于转移矩阵具有行和为 1 的性质，因此每一行只需  $N - 1$  个值即可；而该矩阵一共有  $N$  行，因此需要  $N(N - 1)$  个数；又因为还要定义初始状态下分别取这  $N$  个值的概率，这  $N$  个概率值同样满足和为 1，因此需要  $N - 1$  个数，因此总共需要  $N(N - 1) + (N - 1) = (N + 1)(N - 1) = N^2 - 1$  个数来描述该马尔科夫链。

### 编程部分：

**编程部分：**

利用 HMM 进行中文分词，数据集可自行准备（中文文字不得低于 1000 字）

HMM 的参数为一个五元组，在中文分词里，这五个参数的具体含义如下：

状态值集合为 (B, M, E, S)，观察值集合为所有汉字，这两个参数与三个矩阵通过 viterbi 算法连接起来：观察值集合为 viterbi 算法的输入，状态值集合为 viterbi 算法的输出，在输入和输出之间，三个矩阵则是 viterbi 算法执行需要借助的参数。

相关代码已打包至文件夹内，可以运行、测试。

运行环境：windows+Python3.6

其中 hmm.py 为主程序，实现了用 HMM 进行中文分词；r\_hmm\_data.pkl 则是将词频等信息保存下来以便后续使用；trainCorpus.txt\_utf8 为训练集；testset.txt 为测试集；log.txt 为测试集下的分词结果。（编码格式均为 utf-8）

代码部分简介如下：

```

class HMM(object):
    def __init__(self, load=False):...

    # 将相关信息写入临时文件，方便后续使用，使用时读该文件即可
    def savepara(self):...

    def loadpara(self):...

    # 初始化参数
    def initpara(self, trained=False):...

    def labelmark(self, text):...

    # 从语料中获取词频
    def traincorpus(self, file_path, trained=False):...

    # 词频转化为概率
    def calculate(self):...

    def viterbi(self, text):...

    def cut(self, text, best_path):...

    def use_cut(self, text):...

```

`__init__` ~ `initpara` 都是一些初始化的内容；

`labelmark` 则是给句子中每个词打上标签，如下：

```

def labelmark(self, text):
    length = len(text)
    if length == 1:
        return ['S']
    else:
        return ['B'] + ['M'] * (length - 2) + ['E']

```

`traincorpus` 则是训练过程：取出来字、词，并打上标签，并统计词频；

`calculate` 则是计算状态转移概率、发射概率和初始分布概率；

`viterbi` 则是 `viterbi` 算法实现，课上相关内容不再介绍；

`cut` 和 `use_cut` 则是进行分词（使用 `viterbi` 算法）。

`main` 函数很好理解，就是先用训练集训练，然后用训练好的结果进行测试，将得到的分词结果写入 `log.txt`：

```

if __name__=='__main__':
    f = open('testset.txt', 'r', encoding='utf-8')
    lines = f.readlines()          # 读取测试集数据
    testcase = list()
    for line in lines:
        line = line.strip('\n')
        testcase.append(line)
    print(len(testcase))
    log = open('log.txt', mode='a', encoding='utf-8')
    for i in range(len(testcase)):
        training = HMM(load=True)
        print("第{}行分词结果已写入文件".format(i))
        print(training.use_cut(testcase[i]), file=log)
    # training = HMM(load=True)
    # initmatrix, transprobmatrix, observematrix = training.calculate()
    # print("初始化状态分布结果: {}".format(initmatrix))
    # print("转移概率矩阵: {}".format(transprobmatrix))
    # print("发射概率矩阵: {}".format(observematrix))

```

测试集内有2000行语句，每测试完一行语句会有提示性输出，整个测试过程大约用时1 min。中文分词结果详见文件夹内的 `log.txt`。

在代码文件内main函数被注释掉的最后几行代码为打印三个概率矩阵，可以去掉注释并运行代码得到三个概率矩阵的结果，这里只截取一部分：

```

初始化状态分布结果: {'B': 0.5820129615148393, 'M': 0.0, 'E': 0.0, 'S': 0.4179870384851607}
转移概率矩阵: {'B': {'B': 0.0, 'M': 0.1167175117318146, 'E': 0.8832824882681853, 'S': 0.0}, 'M': {'B': 0.0, 'M': 0.2777743117140081, 'E': 0.7222256882859919, 'S': 0.0}, 'E': {'B': 0.46893265693552616, 'M': 0.0, 'E': 0.0, 'S': 0.5310673430644739}, 'S': {'B': 0.3503213832274479, 'M': 0.0, 'E': 0.0, 'S': 0.46460125869921165}}
发射概率矩阵: {'B': {'中': 0.009227010972739555, '儿': 0.00033416586726125146, '踏': 4.465147364266722e-05, '全': 0.005242227042660882, '各': 0.0035173838269481725, '人': 0.009800998464565456, '共': 0.00215479369578807, '领': 0.003359663299081332, '建': 0.0054294751579365835, '特': 0.0017824580204129253, '社': 0.004439940887210378, '道': 0.000777998634529128, '改': 0.0041878761166469335, '开': 0.00482235915340806, '畜': 0.0002477436602109278, '胜': 0.0002952758740886058, '前': 0.0015901686097259551, '经': 0.007668530505598719, '体': 0.00182638930899684, '纵': 9.506442775535602e-05, '稳': 0.0007036208024013851, '发': 0.00983340672209325, '迈': 9.93855381078722e-05, '步': 0.00025854643609221823, '生': 0.0059163202576534065, '其': 0.0009909746408437111, '事': 0.0022188901660170597, '完': 0.001447571968092921, '七': 0.0006474463678186747, '计': 0.0015944897200784714, '第': 0.0030924746422840814, '任': 0.001428847156565351, '继': 0.0010017774167250016, '有': 0.004612785301311025, '安': 0.001804783757234259, '党': 0.002323316999536201, '风': 0.0007237859840464605, '工': 0.008839551411130605, '取': 0.001103323510009132, '进': 0.0057960493528417065, '民': 0.0031875390700394374, '法': 0.002398936430705234, '不': 0.008148173754728014, '加': 0.0026221937989185703, '十': 0.002584023990804677, '六': 0.0006114371148810398, '通': 0.001951701509219809, '关': 0.0036268519558785824, '精': 0.0017089991444201502, '指': 0.0021756790624918977, '方': 0.004115857610771664, '决': 0.002014357609331294, '文': 0.0037406411951615088, '正': 0.002218169980958307, '水': 0.001868880227463249, '三': 0.0019041692953421312, '实': 0.004262775362757214, '伟': 0.00014331682669178672, '历': 0.0012912918103435858, '转': 0.0007410704254656253, '现': 0.0025782625103346556, '政': 0.006948345446846022, '持': 0.0003096795752636598, '协': 0.0015491180613770514, '己': 0.0010493096306026797, '以': 0.0038760359862070157, '时': 0.0025112852998706546, '之': 0.0022556196040134473, '这': 0.005654172896267425, '一': 0.013713043703710105, '大': 0.006990836365312431, '局': 0.000867822995797, '容': 0.0002196564429195726, '因': 0.0006877767311088257, '我': 0.004099293354420352, '制': 0.002170637767080629, '系': 0.0009657681637873668, '路': 0.0004522762168966938, '符': 0.000244862919975917, '利': 0.0019459408287497874, '愿': 0.0003255236465562191, '客': 0.0004429138111329087, '舰': 0.001949540954043551, '要': 0.0017579717284153336, '根':

```

代码文件内有注释可以帮助阅读整个代码。