

Lab 1: Simple Natural Language Processing

1. Objective

Know how to change the documents to vectors, and make predictions based on the distance.

已知 10 篇训练文本对应的公众“感动”概率值（1、0.9、0.5、0.5、0.4、0.1、0.02、0、0、0），预测 10 篇测试文本对应的公众“感动”概率值各是多少？

2. Dataset

文本编号	词列表（以空格分隔）	公众“感动”的概率
训练文本1	消防员 冲进 火场 救出 男童	1
训练文本2	公务员 患 癌症 保持 在岗	0.9
训练文本3	消防员 多次 冲进 火场 救人 不幸 身亡	0.5
训练文本4	老人 成功 进行 免费 白内障 手术	0.5
训练文本5	海豚 误 吞 排球 后 手术 成功 取出	0.4
训练文本6	6旬 老人 跳楼 自杀 身亡	0.1
训练文本7	男子 跳楼 自杀 身亡	0.02
训练文本8	疑犯 枪杀 出租车 司机	0
训练文本9	男子 枪杀 妻子 后 自杀	0
训练文本10	医师 误 把 肾脏 当 肝脏 致人 身亡	0
测试文本1	癌症 老人 成功 手术	?
测试文本2	男子 枪杀 司机 后 喝药 自杀	?
测试文本3	癌症 医师 保持 手术 清醒	?
测试文本4	男子 跳楼 自杀	?
测试文本5	男子 枪杀 老人 后 自杀	?
测试文本6	消防员 冲进 火场 将 男童 救出	?
测试文本7	出租车 司机 免费 搭载 老人	?
测试文本8	男子 误 杀 弟媳 后 自杀 身亡	?
测试文本9	医师 误 把 患者 肝脏 捅破 致人 身亡	?
测试文本10	6旬 老人 火场 救人 不幸 身亡	?

提供两种格式的数据集：Dataset_txt format.txt、Dataset_excel format.xls

第一种为 txt 格式的数据集（三列之间以 Tab 分隔），供大家方便读取文件中的数据。主要编程语言的参考代码示例如下：

C、C++读写文件：<http://blog.csdn.net/kingstar158/article/details/6859379>

Java 读写文件：<http://blog.csdn.net/jiangxinyu/article/details/7885518>

Matlab 读写文件：<http://blog.csdn.net/yelbosh/article/details/8549121>

Python 读写文件：<http://www.cnblogs.com/allenblogs/archive/2010/09/13/1824842.html>

<http://sucre.iteye.com/blog/704077>

第二种为 excel 格式的数据集，与上述内容一致，供大家参考。

该数据集包含 10 篇训练文本，每篇训练文本既有词列表，也有标准答案（即公众“感动”的概率值）；另有 10 篇测试文本，每篇测试文本只有词列表，其公众“感动”的概率值需要大家预测。数据集很小，供大家开始搭建你们的实验工程，建议选定一种编程语言，后续在此基础上不断完善。后期将有更大数据集。

3. Processes

（1）基于 10 篇训练文本和 10 篇测试文本的词列表，生成不重复的全部词列表文件：输入（Dataset_words.txt）：

文本编号	词列表（以空格分隔）
训练文本1	消防员 冲进 火场 救出 男童
训练文本2	公务员 患 癌症 保持 在岗
训练文本3	消防员 多次 冲进 火场 救人 不幸 身亡
训练文本4	老人 成功 进行 免费 白内障 手术
训练文本5	海豚 误 吞 排球 后 手术 成功 取出
训练文本6	6旬 老人 跳楼 自杀 身亡
训练文本7	男子 跳楼 自杀 身亡
训练文本8	疑犯 枪杀 出租车 司机
训练文本9	男子 枪杀 妻子 后 自杀
训练文本10	医师 误 把 肾脏 当 肝脏 致人 身亡
测试文本1	癌症 老人 成功 手术
测试文本2	男子 枪杀 司机 后 喝药 自杀
测试文本3	癌症 医师 保持 手术 清醒
测试文本4	男子 跳楼 自杀
测试文本5	男子 枪杀 老人 后 自杀
测试文本6	消防员 冲进 火场 将 男童 救出
测试文本7	出租车 司机 免费 搭载 老人
测试文本8	男子 误 杀 弟媳 后 自杀 身亡
测试文本9	医师 误 把 患者 肝脏 捅破 致人 身亡
测试文本10	6旬 老人 火场 救人 不幸 身亡

输出（不重复的全部词列表文件）示例如下：

1	消防员
2	冲进
3	火场
4	救出
5	男童
6	公务员
7	患
8	癌症
9	保持
10	在岗
11	消防员（这个词在第一行出现过，因此不加入不重复的全部词列表文件中）
12	多次
13	冲进（这个词在第二行出现过，因此不加入不重复的全部词列表文件中）
14	火场（这个词在第三行出现过，因此不加入不重复的全部词列表文件中）
15	救人
16

（2）基于上一步骤输出的不重复的全部词列表文件，将原始的 10 篇训练文本和 10 篇测试文本都转换为向量，如下所示：

文本编号	消防员	冲进	火场	救出	男童	公务员	患	癌症	保持	在岗	多次	救人
训练文本1	1	1	1	1	1	0	0	0	0	0	0	0
训练文本2	0	0	0	0	0	1	1	1	1	1	0	0
训练文本3	1	1	1	0	0	0	0	0	0	0	1	1
.....													
测试文本1	0	0	0	0	0	0	0	1	0	0	0	0
.....													

其中，第一行中的“消防员”、“冲进”等词，为上一步骤输出的不重复的全部词，以空格分隔（也可以用逗号、分号、Tab 等分隔，程序能够处理即可）；第二行中的“1”、“1”等值，获得的方式为：如果当前文本中含有“消防员”这个词，则“消防员”这一列的值即为“1”，否则为“0”。比如，训练文本 1 是“消防员 冲进 火场 救出 男童”，所以训练文本 1 的“消防员”这一列的值为“1”。训练文本 2 是“公务员 患 癌症 保持 在岗”，所以训练文本 2 的“消防员”这一列的值为“0”。据此，原始的 10 篇训练文本和 10 篇测试文本都被转换为向量，向量中各个维度上的值要么是 1，要么是 0。如下：

训练文本 1 = (1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,)

训练文本 2 = (0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0,)

.....

测试文本 1 = (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,)

.....

（3）基于上一步骤输出的向量文件，采用下述方法预测 10 篇测试文本对应的公众“感动”的概率值：

假如现在要预测测试文本 1 对应的公众“感动”的概率值，首先计算测试文本 1 的向量与 10 篇训练文本（即训练文本 1、训练文本 2、...、训练文本 10）的向量的欧式距离。这里再回顾一下欧式距离的计算公式。6 维向量(1, 1, 1, 1, 0, 0)与 6 维向量(0, 1, 0, 0, 1, 1)的欧式距离为： $\sqrt{(1-0)^2 + (1-1)^2 + (1-0)^2 + (1-0)^2 + (0-1)^2 + (0-1)^2}$ ，即两个向量在每一个维度上的值的差的平方和，再开根号。然后，上一个步骤，我们会得到测试文本 1 的向量分别与训练文本 1 的向量、训练文本 2 的向量、...、训练文本 10 的向量的 10 个欧式距离的值，比较这 10 个欧式距离的值，得到一个最小的欧式距离的值。假如通过比较得到测试文本 1 的向量与训练文本 4 的向量的欧式距离最小，则将测试文本 1 对应的公众“感动”的概率值预测为训练文本 4 对应的公众“感动”的概率值，即 0.5。采用上述相同的方法，得到测试文本 2 至测试文本 10 对应的公众“感动”的概率值的预测值，比如分别是：0.1、0.4、0、1、0.9、0.02、1、0.5、0。

(4) 运行 RunResult.bat 文件（请将 AILab.jar 与该文件放在同一目录下，若仍无法运行该文件，请在此下载 jre 后安装：<http://www.java.com/en/download/manual.jsp>），输入上一步骤获得的 10 篇测试文本对应的公众“感动”的概率值的预测值，得到相关系数的值，如下：

```
请依次输入10篇测试文本的预测结果，以空格或回车分隔：
0.5 0.1 0.4 0 1 0.9 0.02 1 0.5 0
你的上述预测结果与标准答案的相关系数(-1到1之间)为：
0.046830853318700745
```

(5) 以自己的学号新建一个文本文件，如 99999999.txt，将“exp1、空格、以及上述相关系数的输出值（保留小数点后四位，即 0.0468）”写入该文本文件中的第一行：

```
99999999.txt
1 exp1 0.0468
```

4. More experiments

在实现了“3. Process”中的全部内容后，请大家继续下述扩展实验：

4.1 更改向量中值的表示方法

在“3. Process”的（2）步骤中，每篇文本转换成的向量中，各个维度上的值要么是 1，要么是 0，如下：

文本编号	消防员	冲进	火场	救出	男童	公务员	患	癌症	保持	在岗	多次	救人
训练文本1	1	1	1	1	1	0	0	0	0	0	0	0
训练文本2	0	0	0	0	0	1	1	1	1	1	0	0
训练文本3	1	1	1	0	0	0	0	0	0	0	1	1
.....													
测试文本1	0	0	0	0	0	0	0	1	0	0	0	0
.....													

现在，请将上述向量中的值归一化，即对于每篇文本对应的向量，都将其每个维度上的值同时除以这篇文本的总词数（或者说这篇文本对应向量的所有维度上的值的和），比如，在“3. Process”的（2）步骤中，训练文本 1 = (1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,), 这个向量的所有维度上的值的和为 5（因为训练文本 1 总共包含 5 个词），那么，归一化以后的训练文本 1 对应的向量 = (1/5, 1/5, 1/5, 1/5, 1/5, 0, 0, 0, 0, 0, 0, 0, 0,)= (0.2, 0.2, 0.2, 0.2, 0.2, 0, 0, 0, 0, 0, 0, 0,). 按照上述方法，将全部的 10 篇训练文本和 10 篇测试文本对应的向量都进行归一化，然后按照“3. Process”的后续步骤，将“exp2、空格、以及在归一化的向量文件中进行预测输出的相关系数的值（保留小数点后四位）”继续写入 99999999.txt 的第二行：

```
99999999.txt
1 exp1 0.0468
2 exp2 0.0242
```

【注意】上述 0.0468、0.0242 仅供举例，你们运行得到的值应该不会与之相同。另外**请保存上述过程中所有的中间文件**，后期实验中可能会用到。

4.2 更多的方法

在“3. Process”的（3）步骤中，我们采用最小的欧式距离作为预测的公众“感动”的概率值，如果采用最小的 City block distance，或者最小的 Supremum distance 作为概率预测的基准，相关系数会如何变化？如果不是只采用最小的欧式距离对应的训练文本的公众“感动”的概率值作为预测值，而是将每个欧式距离的倒数等作为权重（由于欧式距离的倒数可能会是一个很大的正数，此时可能需要采用归一化的方法将其转换为(0, 1]之间）加权计算，相关系数又会如何变化？

归一化方法参考：

[1] http://baike.baidu.com/link?url=3dqGpByo1AIVQD008bNUeNYv9WtJvKa1cZhvv7rz147rhOhbCn_I84owwcgcphLHnTXAGVbt1Q4q-UiMs6j6K

[2] <http://www.cnblogs.com/chaosimple/archive/2013/07/31/3227271.html>

请在时间允许的前提下，开拓思路进行实验，并将所有实验当中，输出的相关系数最大的那个值，以“exp3、空格、以及最大的相关系数的值（保留小数点后四位）”的格式继续写入 99999999.txt 的第三行，比如：

```
99999999.txt
1 exp1 0.0468
2 exp2 0.0242
3 exp3 0.8000
```

以自己的学号新建一个 word（或 PDF）文件，如 99999999.doc，word（或 PDF）文件中将“4.2 更多的方法”中你的实验方法，以及关键代码（包括“3. Process”以及“4.1 更改向量中值的表示方法”中的实现代码）简要地描述或截图出来即可。本次实验历时两周，即 2015 年 10 月 15 日、16 日，以及 2015 年 10 月 22 日、23 日，时间还算充裕。但请大家在每次实验课结束前，提交下述文件（需要提交 2 次，2015 年 10 月 15 日、16 日的本周实验课，提交 1 次；2015 年 10 月 22 日、23 日的下周实验课，提交 1 次。每次提交的进展，根据每位同学的基础会有不同，但本周实验课提交的文件中，至少应该实现了“3. Process”中的核心代码）：

【1】将 99999999（**改为你自己的学号**）.txt 上传到 Lab 1 results 目录中；

【2】将 99999999（**改为你自己的学号**）.doc 或 99999999.pdf 上传到 Lab1 reports 目录中。

FTP 服务器：ftp://smie2.sysu.edu.cn

帐号：student0007

密码：student0007

【注意】提交上述 2 个文件前，请再三确认自己的学号无误！