1. 清洗数据

对food.txt数据进行清洗，去掉该文件中的所有非中文字符，每一行只能放一个中文词汇

保存文件名为：姓名\_new\_food\_1.txt ，处理代码文件名为：姓名\_process\_1.py

2. 挑选与去重

去掉 new\_food1.txt 中所有非食物词汇的单词，并去重，请保留原本词汇的相对顺序

保存文件名为：姓名\_new\_food\_2.txt ，处理代码文件名为：姓名\_process\_2.py

3. 使用热门模型进行编码

使用 DeepSeek-R1-Distill-Qwen-1.5B 中的 Embedding，对 姓名\_new\_food\_2.txt 中所有词汇进行词向量编码

文件保存为：姓名\_new\_food\_3.txt，处理代码文件名为：姓名\_process\_3.py

姓名\_new\_food\_3.txt 为可直接打开的文件，格式如下：

【桂皮】：[embedding1]

【蒜】：[embedding2]

......

4. 查找相似词汇

求出 姓名\_new\_food\_3.txt 中和 【苹果】、【香蕉】、【梨子】、【西瓜】、【荔枝】、【桂圆】、【橘子】、【布丁】分别最相近的100个词，共计808个词汇（自身也算）。

文件保存为：姓名\_new\_food\_4.txt，处理代码文件名为：姓名\_process\_4.py

姓名\_new\_food\_4.txt 中的文件格式为：

【苹果】：词汇1、词汇2、词汇3、......、【词汇99】、【词汇10】

【香蕉】：词汇1、词汇2、词汇3、......、【词汇99】、【词汇10】

.....

5. 空间映射

加载 姓名\_new\_food\_4.txt ，使用线性层对它们进行二维空间映射，并使用 matplotlib 将该二维空间画出来。

文件保存为：姓名\_new\_food\_5.png，处理代码文件名为：姓名\_process\_5.py

6. 个性化训练

对 【苹果】、【香蕉】、【梨子】、【西瓜】、【荔枝】、【桂圆】、【橘子】、【布丁】 最相近的100个词汇进行标注，并完成个性化训练，使得本身最相近的词汇，变成最不相近。

例如：【苹果】原本最相近的词汇为 【词汇1】、【词汇2】、......、【词汇99】、【词汇100】，经过标注和训练后，【苹果】最相近的词汇为 【词汇100】、【词汇99】、......、【词汇2】、【词汇1】

文件保存为：姓名\_new\_food\_6.txt，处理代码文件名为：姓名\_process\_6.py

姓名\_new\_food\_6.txt 中的文件格式为：

【苹果】：词汇1、词汇2、词汇3、......、【词汇99】、【词汇10】

【香蕉】：词汇1、词汇2、词汇3、......、【词汇99】、【词汇10】

......