



## 语言分析与机器翻译报告

题目：使用 NMT 德译英+使用 RNN 训练语言模型-tensorflow

学院：计算机科学与工程学院

专业：计算机科学与技术

姓名：石玲玲

学号：2001801

授课教师：朱靖波、肖桐

完成日期：2020 年 12 月 15 日

## 目录

<b>一、使用 NiuTrans.NMT 进行 Translating 实践.....</b>	<b>3</b>
1、实验目的.....	3
2、实验过程： .....	3
3、实验结果.....	3
4、实验结果分析.....	4
<b>二、使用 RNN 训练语言模型—tensorflow.....</b>	<b>5</b>
1、实验目的.....	5
2、实验预期.....	5
3、实验原理.....	5
4、实验过程.....	5
5、实验结果与分析.....	6
<b>三、选修《语言分析与机器翻译》的心得体会.....</b>	<b>15</b>

## 一、使用 NiuTrans.NMT 进行 Translating 实践

### 1、实验目的

体验 NiuTrans.NMT

### 2、实验过程:

①在 NiuTrans.NMT 中找到 Translating 的 example

②在 example 最下方找到模型链接，本次实验使用的模型为 iwslt14.en-de.ensemble;

③在 NMT 压缩包中找到训练数据集，本次实验使用的数据集为 iwslt14.tokenized.de-en 中的 test.de

④根据说明改变参数，本次实验最终成功运行的参数为:

```
-dev -1 -test  
C:\Users\lenovo-pc\GitHub\NiuTrans.NMT-master\sample\train\iwslt14.tokenized.d  
e-en\iwslt14.tokenized.de-en\test.de -model E:\硕士研究生\机器翻译\机器翻译  
\iwslt14.de-en.ensemble\iwslt14.en-de.ensemble\model.bin -sbatch 64  
-beamsize 1 -srcvocab E:\硕士研究生\机器翻译\机器翻译  
\iwslt14.de-en.ensemble\iwslt14.en-de.ensemble\vocab.de -tgtvocab E:\硕士研究  
生\机器翻译\机器翻译\iwslt14.de-en.ensemble\iwslt14.en-de.ensemble\vocab.en  
-output output.atat
```

⑤在 NiuTrans.NMT 中输入上面的参数并执行

### 3、实验结果

在参数设置的对应路径中生成 output 文件

我的电脑 > Windows (C:) > 用户 > lenovo-pc > NMT\_class >

名称	修改日期	类型	大小
.vs	2020/11/17 21:45	文件夹	
CMakeFiles	2020/11/17 21:46	文件夹	
NiuTrans.NMT.dir	2020/11/17 21:45	文件夹	
ALL_BUILD.vcxproj	2020/11/17 21:44	VC++ Project	48 KB
ALL_BUILD.vcxproj.filters	2020/11/17 21:44	VC++ Project Fil...	1 KB
cmake_install.cmake	2020/11/17 21:44	CMAKE 文件	2 KB
CMakeCache.txt	2020/11/17 21:44	文本文档	14 KB
NiuTrans.NMT.sln	2020/11/17 21:45	Visual Studio Sol...	2 KB
NiuTrans.NMT.vcxproj	2020/11/17 21:44	VC++ Project	98 KB
NiuTrans.NMT.vcxproj.filters	2020/11/17 21:44	VC++ Project Fil...	69 KB
NiuTrans.NMT.vcxproj.user	2020/11/17 21:56	Per-User Project...	1 KB
output.atat	2020/11/17 21:59	ATAT 文件	1 KB
ZERO_CHECK.vcxproj	2020/11/17 21:44	VC++ Project	48 KB
ZERO_CHECK.vcxproj.filters	2020/11/17 21:44	VC++ Project Fil...	1 KB

下面是输出结果展示

\*output.atat - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

you know , one of the great minds of travel@@ ing and one of the pleas@@ ures about eth@@ no@@ graph@@ y research is to live with the people who still remember the old days . they still feel their past in the wind , they touch them on the ra@@ ins that were gl@@ ori@@ fied , they taste in the bit@@ ter leaves of the plants .

上述翻译的原文:

\*test.de - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

wissen sie , eines der großen vern@@ ü@@ gen beim reisen und eine der freu@@ den bei der eth@@ no@@ graph@@ ischen forschung ist , gemeinsam mit den menschen zu leben , die sich noch an die alten tage erinnern können . die ihre vergangenheit noch im mer im wind spüren , sie auf vom regen ge@@ gl@@ ä@@ t@@ teten st@@ einen berü@@ hren , sie in den bit@@ teren blä@@ ttern der pflanzen schme@@ cken .

#### 4、实验结果分析

重现了 NMT 里对 Translating 的样例，即实现了将给定数据集中的德语翻译成英语；但是未能对模型进一步优化，仅仅只是重现，而且并没有对实验结果进行评测。

## 二、使用 RNN 训练语言模型—tensorflow

### 1、实验目的

进一步深刻体验 NLP

### 2、实验预期

通过 RNN 网络对一段文字的训练学习生成模型，最终实现输入某些文字到模型里，模型自动预测后面的文字，同时将模型预测出来的文字当成输入，再放到模型里，模型就会预测出下一个文字，这样循环下去，可以看到 RNN 能够输出一句话，即生成与这段文字相关的文字并尽可能地语句通顺。

### 3、实验原理

将整段文字都看成一个个的序列。在模型里预设值只关注连续的 4 个序列，在整段文字中，每次随机取 4 个连续的文字放到模型里进行训练，然后把第 5 个连续的值当成标签，与输出的预测值进行 loss 的计算，形成一个可训练的模型，通过优化器来迭代训练。

### 4、实验过程：

#### ①准备样本

本次实验共使用了四组样本分别训练模型。通过代码实现读取整体样本，获取全部的字表 words，并生成样本向量 wordlabel 和与向量对应关系的 word\_num\_map。

#### ②构建模型

本次实验中使用 3 层的 LSTM RNN 模型，第一层为 256 个 cell，第二层和第三层都是 512 个 cell，将 x 形状变换并按时间序列裁分，然后放入 3 层 LSTM 网络，最终通过一个全连接生成 words\_size 个节点，后面接入一个 softmax 分类，对下一个字属于哪个向量进行分类，这里认为一个字就是一类。

学习率为 0.001，迭代 20000 次，每 500 次输出一次中间状态。每次输入 4 个字，来预测第 5 个字。

#### ③定义优化器

本次实验中使用的优化器是 AdamOptimizer，使用的 loss 是 softmax 的交叉熵，正确率是统计 one\_hot 中索引对应的位置相同的个数。

#### ④训练模型

在训练过程中添加保存检查点功能。在 `session` 中每次随机取一个偏移量，然后取后面 4 个文字向量当作输入，第 5 个文字向量当作标签用来计算 `loss`。

### ⑤使用模型进行预测

启用一个循环，等待输入文字，当收到输入的文本后，通过 `eval` 计算 `onehot_pred` 节点，并进行文字的转义，得到预测文字。接下来将预测文字再循环输入模型中，预测下一个文字。代码中设定循环 32 次，输出 32 个文字。

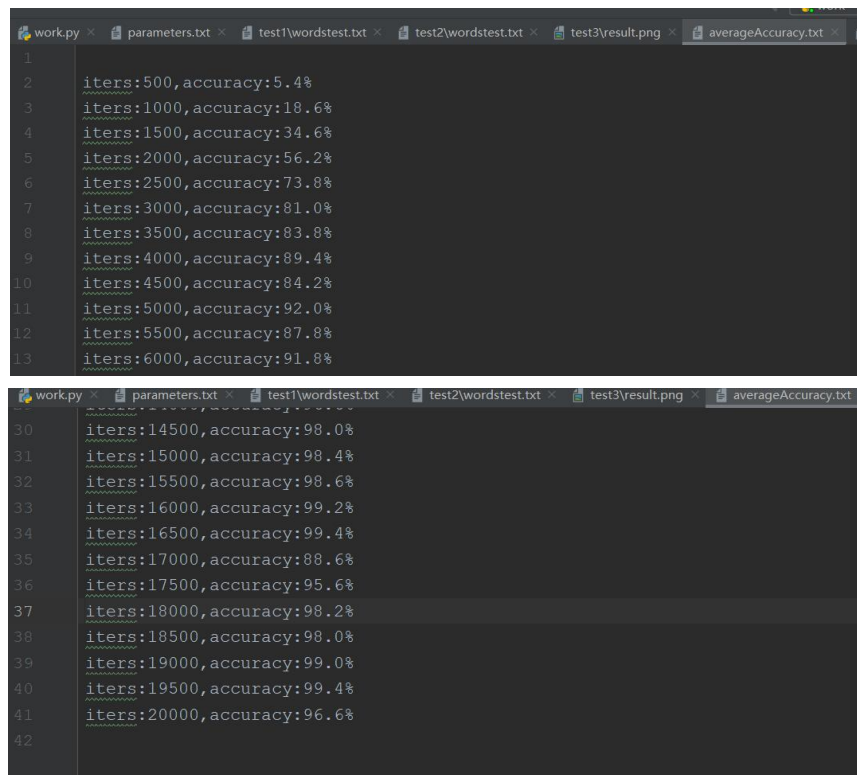
本次实验中，使用训练好的模型进行了大量的预测，下面进行实验结果分析。

## 5、实验结果与分析

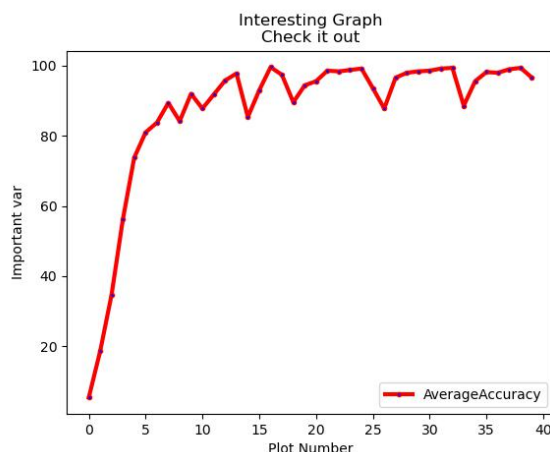
### ①模型一的实验结果与分析

**使用的样本为：**在尘世的纷扰中，只要心头悬挂着远方的灯光，我们会坚持不懈地走，理想为我们灌注了精神的蕴藉。所以，生活再平凡、再普通、再琐碎，我们都要坚持一种信念，默守一种精神，为自己积淀站立信心，前行的气力。

在使用该样本**对模型进行训练**时，迭代次数设置为 20000 次，每迭代 500 次记录一次准确率。下面是随着迭代次数的增加，**准确率的部分变化过程**：

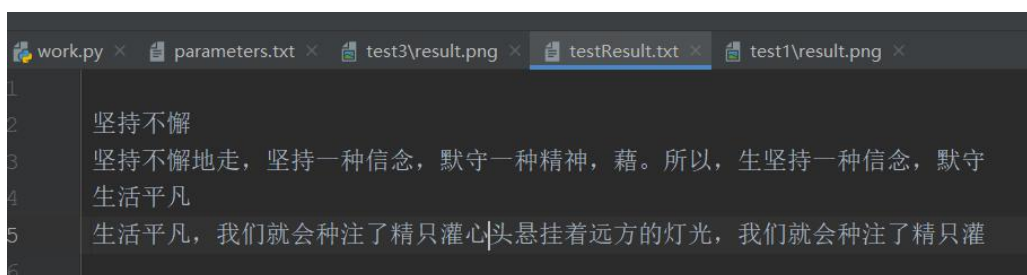


为了直观表示准确率的走向，本次实验绘制了**准确率走势的折线图**：



使用本样本对模型训练，在迭代 20000 次准确率达到 96.6%，认为模型可用，开始使用该模型进行预测。

本次实验中使用该模型进行了两次预测，向训练好的模型输入文字“坚持不懈”和“生活平凡”，模型相应的预测结果为：



从预测结果看，模型准确率相对较高，但是模型不太稳定，因此在对文字进行预测时，虽然大致能看明白输出的语句是有一定逻辑的，但是还是不是完美的语句。

## ②模型二的实验结果与分析：

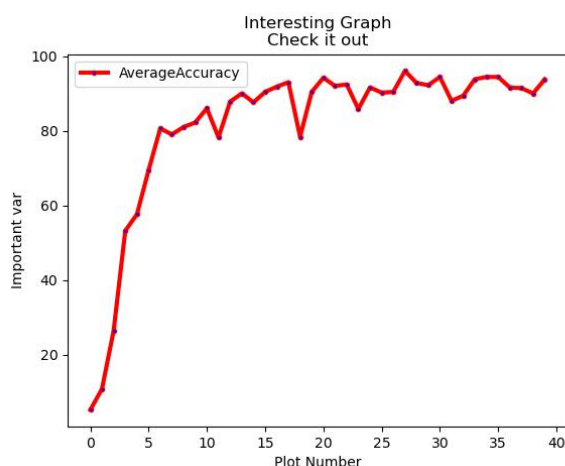
**使用的样本为：**这里将整段文字都看成一个个的序列。在模型里预设值只关注连续的 4 个序列，这样在整段文字中，每次随意拿出 4 个连续的文字放到模型里进行训练，然后把第 5 个连续的值当成标签，与输出的预测值进行 loss 的计算，形成一个可训练的模型，通过优化器来迭代训练。

在使用该样本对模型进行训练时，迭代次数设置为 20000 次，每迭代 500 次记录一次准确率的值。下面是随着迭代次数的增加，准确率的部分变化过程：

```
work.py x parameters.txt x test3\result.png x averageAccuracy.txt x test1\result.png x
1  iters:500,accuracy:5.2%
2  iters:1000,accuracy:10.8%
3  iters:1500,accuracy:26.4%
4  iters:2000,accuracy:53.2%
5  iters:2500,accuracy:57.6%
6  iters:3000,accuracy:69.4%
7  iters:3500,accuracy:80.6%
8  iters:4000,accuracy:79.0%
9  iters:4500,accuracy:81.0%
10 iters:5000,accuracy:82.2%
11 iters:5500,accuracy:86.0%
12 iters:6000,accuracy:78.2%
13 iters:6500,accuracy:87.8%
14 iters:7000,accuracy:90.0%
15 iters:7500,accuracy:87.6%

29 iters:14500,accuracy:92.8%
30 iters:15000,accuracy:92.2%
31 iters:15500,accuracy:94.4%
32 iters:16000,accuracy:88.0%
33 iters:16500,accuracy:89.4%
34 iters:17000,accuracy:93.8%
35 iters:17500,accuracy:94.4%
36 iters:18000,accuracy:94.4%
37 iters:18500,accuracy:91.6%
38 iters:19000,accuracy:91.4%
39 iters:19500,accuracy:90.0%
40 iters:20000,accuracy:93.8%
41
```

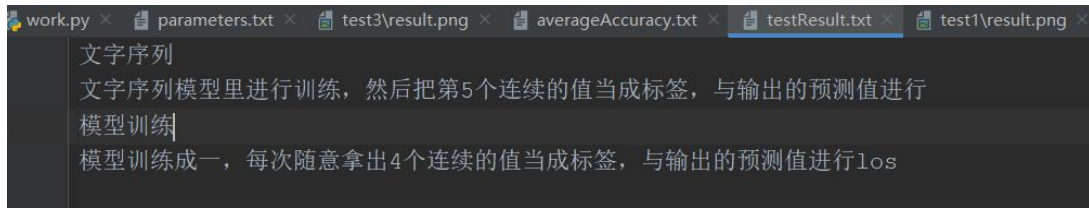
为了直观表示准确率的走向，本次实验绘制了准确率走势的折线图：



使用本样本对模型训练，在迭代 **20000** 次准确率达到 **93.8%**，认为模型可用，开始使用该模型进行预测。

本次实验中使用该模型进行了两次预测，向训练好的模型输入文字“文字序列”和“模型训练”，模型相应的预测结果为：



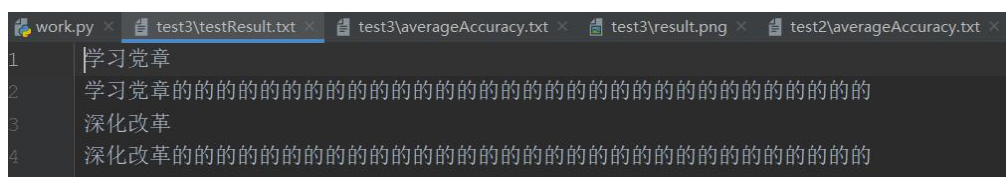


从预测结果看，模型准确率相对较高，但是模型不太稳定，因此在对文字进行预测时，虽然大致能看明白输出的语句是有一定逻辑的，但是还是不是完美的语句。但是与模型一相比，虽然准确率低，但是文字预测出的结果更接近人类语言。

### ③模型三的实验结果与分析：

**使用的样本为：**在思想上。在这段时间来，我再一次认真系统的学习了新党章、“三个代表”重要思想和科学发展观;深刻领会“十九大”会议精神，并充分认识到它们是改造客观世界，夺取社会主义现代化建设事业胜利果实的行动指南。经过这一系列的学习，我不断提高自我的政治思想水平，更加坚定了加入中国共产党的信念，并且懂得了理论上的成熟是政治上成熟的基础，政治上的清醒于稳固的理论基石。仅有坚定了共产主义信念，牢记全心全意为人民服务的宗旨，才能在这个风云变幻的国际环境中，在深化改革、扩大开放、大力发展市场经济的全新形势下，始终坚持党的基本路线不动摇，永远坚持一个党员应有的纯洁性和先进性。在工作上。作为一名入党进取分子，我时刻都严格要求自我，努力工作，不等不靠，在工作中我严格以党员的标准来要求自我，牢记入党誓词，克服并纠正自身存在的问题，工作中大胆负责，遇到困难挺身而出。牢记党对我的培养和教育，吃苦在前，享受在后，能够脚踏实地任劳任怨的工作，并能够根据实际情景合理做好前后保障工作，为我系的工作尽职尽责!另外在做好本职工作的同时能够虚心学习其它各相关专业，力求做到一专多能，以更好地为师生服务。在生活中。我认为:为师生服务不仅仅能够体此刻大事上，的是体此刻平常的一些细节上，平时不能不屑于做小事而在等做大事，人有云:一屋不扫何以扫天下所以要从小事做起，从身边做起。日常的生活中，我一向都以一个党员的标准严格要求自我，遵守国家法律法规，遵守社会公德，发挥党员的模范带头作用，进取团结同事，热心助人，主动帮忙其他同事做一些力所能及的事。作为在党的关心和培养下成长起来的消防员，单有一腔热血和为人民服务的热情是远远不够的，还需要有坚实的科学文化知识作为基础，所以，我进取的利用业余时间学习，

在使用该样本**对模型进行训练**时，迭代次数设置为 20000 次，每迭代 500 次记录一次准确率的值。下图是**迭代 20000 次时准确率**的折线图以及预测结果（经过实践，该样本下**迭代 20000 次**的最终准确率只有 **3.8%**，效果太差，模型不可用，因此，使用这个不好的模型进行预测时，得到的是没有意义的语句）：

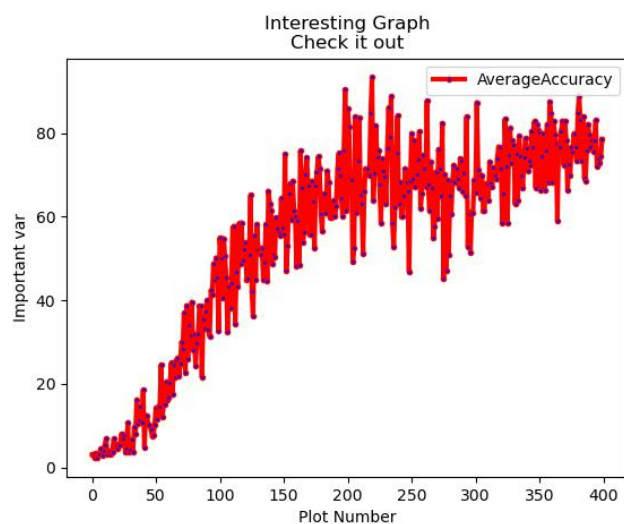


下面是迭代次数为 200000 次，每 500 次记录一次准确率，随着迭代次数的增加，准确率的部分变化过程：

```
work.py x averageAccuracy.txt x parameters.txt x result.png x test3\testResult.txt x wordtest.txt
iters:500,accuracy:3.0%
~~~~~
iters:1000,accuracy:3.0%
~~~~~
iters:1500,accuracy:2.2%
~~~~~
iters:2000,accuracy:3.4%
~~~~~
iters:2500,accuracy:2.4%
~~~~~
iters:3000,accuracy:3.2%
~~~~~
iters:3500,accuracy:3.2%
~~~~~
iters:4000,accuracy:4.4%
~~~~~
iters:4500,accuracy:2.8%
~~~~~
iters:5000,accuracy:4.0%
~~~~~
iters:5500,accuracy:5.4%
~~~~~
iters:6000,accuracy:6.8%
~~~~~
iters:6500,accuracy:3.2%
~~~~~
iters:7000,accuracy:4.0%
~~~~~
iters:7500,accuracy:3.6%
~~~~~
iters:8000,accuracy:3.2%
~~~~~

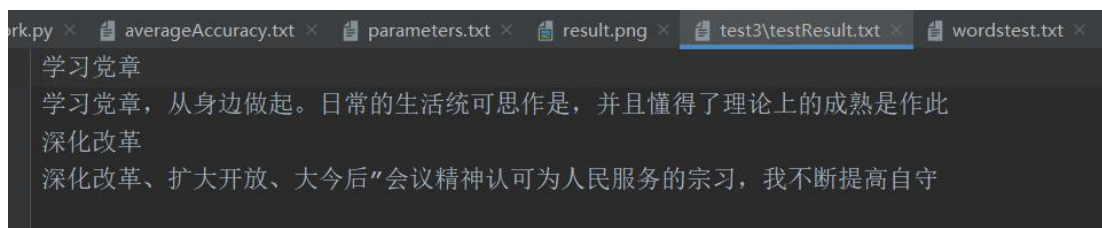
work.py x averageAccuracy.txt x parameters.txt x result.png x test3\testResult.txt x wordtest.txt x
10 iters:193000,accuracy:76.0%
11 iters:194000,accuracy:82.2%
12 iters:195000,accuracy:79.8%
13 iters:195500,accuracy:76.6%
14 iters:196000,accuracy:78.4%
15 iters:196500,accuracy:75.6%
16 iters:197000,accuracy:78.4%
17 iters:197500,accuracy:83.2%
18 iters:198000,accuracy:72.0%
19 iters:198500,accuracy:74.0%
20 iters:199000,accuracy:72.8%
21 iters:199500,accuracy:74.4%
22 iters:200000,accuracy:78.6%
23
```

下图是迭代 200000 次时模型对应的准确率的折线图：



重新对模型进行训练，在迭代 200000 次准确率达到了 78.6%，认为模型可用，开始使用该模型进行预测。

本次实验中使用该模型进行了**两次预测**，向训练好的模型输入文字“深化改革”和“学习党章”，模型相应的**预测结果**为：



The screenshot shows a text editor with several tabs open: 'rk.py', 'averageAccuracy.txt', 'parameters.txt', 'result.png', 'test3\testResult.txt', and 'wordstest.txt'. The active tab is 'test3\testResult.txt', which contains the following text:

```
学习党章  
学习党章，从身边做起。日常的生活统可思作是，并且懂得了理论上的成熟是作此  
深化改革  
深化改革、扩大开放、大今后“会议精神认可为人民服务的宗习，我不断提高自守
```

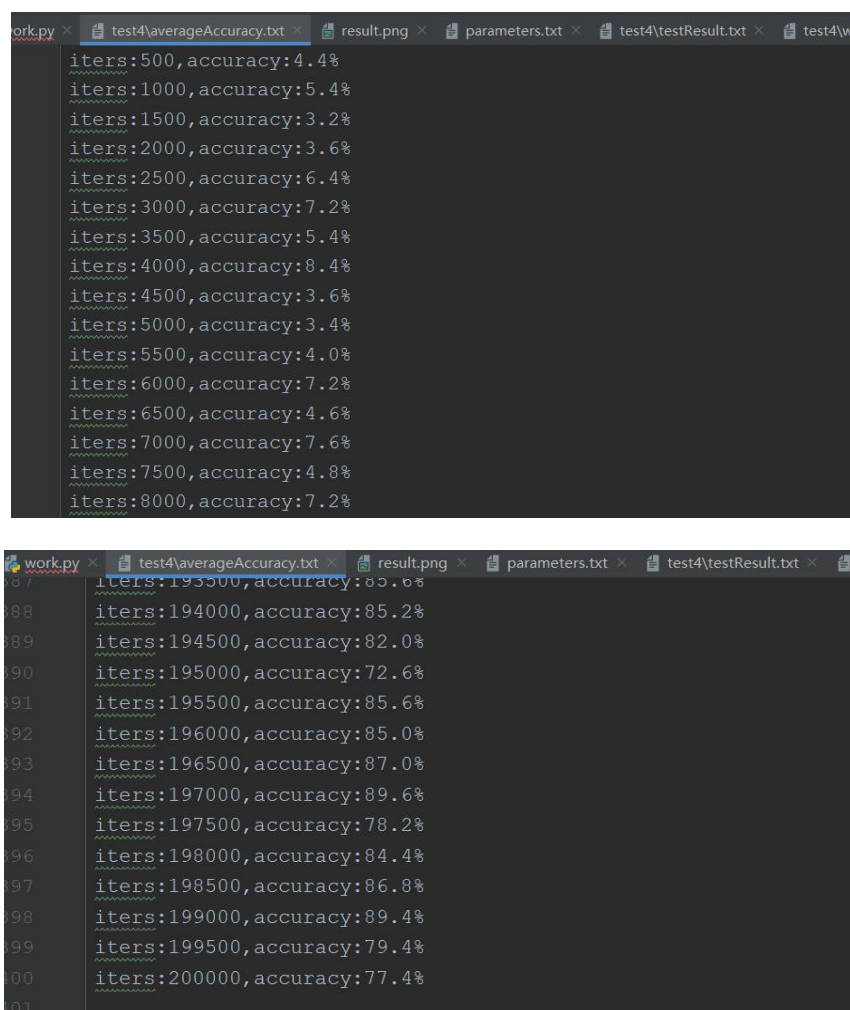
理论上讲，随着样本的增大，模型的准确率和稳定性较前两个模型应该有大幅度的提升，但是由于迭代次数没有足够大，没有将模型尽可能地训练好。考虑到现实因素，本次实验对于该模型只训练了两次，分别是迭代次数为 20000 次和 200000 次。对比来看，模型训练时迭代次数越多，模型的准确率越好，但是模型的性能还有较大的提升，考虑到现实因素，没有进一步对模型进行优化。

#### ④模型四的实验结果与分析

**使用的样本为：**辣椒引进四川进行种植并广泛运用于川菜烹调中，是古代川菜与近代川菜划分的一个分水岭，被视为近代川菜初现雏形的开始，这个时期大致在清朝初期的康熙时代。康熙二十七年（公元 1688 年）陈溴子撰写出版的《花镜》一书在第五卷有记载：“番椒，一名海疯藤，俗名辣茄……其味最辣，人多采用，研极细，冬月取以代胡椒。”这里的番椒，就是辣椒，也称海椒、秦椒等。而辣椒与蚕豆（即胡豆）的完美结合创制出的被誉为川菜灵魂的四川豆瓣被广泛运用于川菜烹调中，则被视为近代川菜形成的标志。豆瓣，俗称胡豆瓣，在品种繁多的四川豆瓣中，以郫县豆瓣最为著名。继而泡椒、泡菜、豆豉在川菜烹调中的革新运用，以及川菜三大类 24 种常用味型、54 种烹调方法和 3000 余款经典传统名菜的形成，是近代川菜最终成型并成为中国四大菜系之首的标志，这个时间在民国中后期。两宋四川饮食的重大成就，就在于其烹饪开始被送到境外，让境外的川人和不是川人的普通人能在专门的食店里吃到具有地方特色的风味饮食，这是四川菜第一次成为一个独立的烹调体系的伊始。这就是所谓北宋的“川饭”，这些川饭店，主要经销“插肉面、大爰面、大小抹肉淘、煎爰肉、杂煎事件、生熟烧饭。”南宋的“川饭分茶”。从上述两书的内容可以发现，川菜出川主要经营大众化的饮食，尤其是面食，而面食里占主要成分的品种是面条，附带也有一些快餐类肉食。今日上海、杭州面条里的“爰面”或“沃面”很可能是川

饭面条的遗存，因为我们在《东京梦华录》（写于南宋初年）里找不到第二处有记载爇面的地方，根据《都城纪胜·食店》，南渡以后的南食店和川饭分茶事实上成了面食店的代称，因此北宋开封川饭店的爇面在南渡一百五十年以后很可能变成一种固定的江南面条了。而我们知道，现代的爇面已经和现代川菜面条大不一样了。这些烹调的具体调味特色，而且没有发现其厚味、辛香的特色。从《梦粱录》的说明中，我们知道川饭的出现原因是，在北宋时期，为照顾在汴京居住的蜀中士大夫的口味，“谓其不便北食故耳。”南渡一百五十年以后，这些随南渡开设到临安的川饭店，已经“无南北之分矣”，说明这些川味面食曾与中原烹调有较大差异。

在使用该样本**对模型进行训练**时，迭代次数设置为 200000 次，每迭代 500 次记录一次准确率的值。下面是随着迭代次数的增加，**准确率的部分变化过程**：

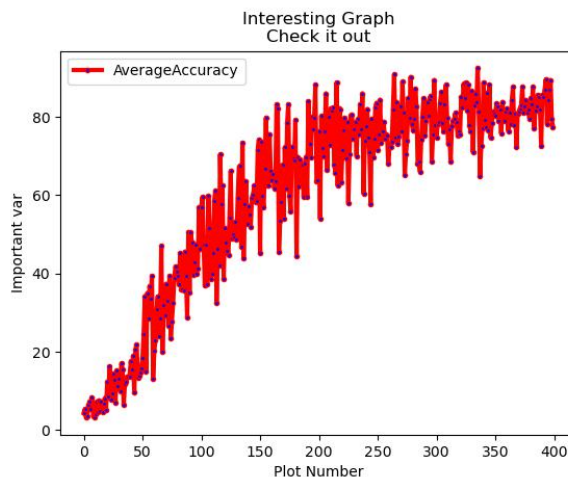


```
ork.py x test4\averageAccuracy.txt x result.png x parameters.txt x test4\testResult.txt x test4\w
iters:500,accuracy:4.4%
~~~~~
iters:1000,accuracy:5.4%
~~~~~
iters:1500,accuracy:3.2%
~~~~~
iters:2000,accuracy:3.6%
~~~~~
iters:2500,accuracy:6.4%
~~~~~
iters:3000,accuracy:7.2%
~~~~~
iters:3500,accuracy:5.4%
~~~~~
iters:4000,accuracy:8.4%
~~~~~
iters:4500,accuracy:3.6%
~~~~~
iters:5000,accuracy:3.4%
~~~~~
iters:5500,accuracy:4.0%
~~~~~
iters:6000,accuracy:7.2%
~~~~~
iters:6500,accuracy:4.6%
~~~~~
iters:7000,accuracy:7.6%
~~~~~
iters:7500,accuracy:4.8%
~~~~~
iters:8000,accuracy:7.2%

ork.py x test4\averageAccuracy.txt x result.png x parameters.txt x test4\testResult.txt x t
887 iters:193500,accuracy:85.6%
888 iters:194000,accuracy:85.2%
889 iters:194500,accuracy:82.0%
890 iters:195000,accuracy:72.6%
891 iters:195500,accuracy:85.6%
892 iters:196000,accuracy:85.0%
893 iters:196500,accuracy:87.0%
894 iters:197000,accuracy:89.6%
895 iters:197500,accuracy:78.2%
896 iters:198000,accuracy:84.4%
897 iters:198500,accuracy:86.8%
898 iters:199000,accuracy:89.4%
899 iters:199500,accuracy:79.4%
900 iters:200000,accuracy:77.4%
901
```

为了直观表示准确率的走向，本次实验绘制了**准确率走势的折线图**：





使用本样本对模型训练，在迭代 **200000** 次准确率达到 **77.4%**，认为模型可用，开始使用该模型进行预测。

本次实验中使用该模型进行了两次预测，向训练好的模型输入文字“川菜形成”和“四大菜系”，模型相应的预测结果为：

```
work.py x test4\averageAccuracy.txt x result.png x parameters.txt x test4\testResult.txt x test4\wordstest.txt
1 川菜形成
2 川菜形成的标志。的说》一书在第五卷有南能是面条，的，以川菜初现雏形的运用
3 四大菜系
4 四大菜系之首的营大众化现，食：的子方时色北宋豆划味主、样了。这些烹调，现
5
```

从预测结果看，模型准确率不是很高，而且模型不太稳定，因此在对文字进行预测时，预测结果不太好，但是与模型三对比来看，两个模型的训练样本差不多大，迭代相同次数后的准确率也差不多，因此可以通过加大迭代次数进一步优化模型从而得到更好的预测结果，考虑到现实因素，本次实验没有进一步优化。

### 三、选修《语言分析与机器翻译》的心得体会

首先非常感谢肖桐老师在研究生的课堂里提供了上机练习的机会，这是我本学期所有课程中唯一需要自己真正动手操作的课程。正是因为有了这样的要求，我清醒且深刻地认识到自己的短板有多短，因此很感谢肖桐老师通过这样的方式让我对自己有了更加清晰的认识，同时找到了自己以后的努力目标。除去学习，肖桐老师是我所修课程中一个坚持本真的老师，比如肖桐老师会喝奶茶，而且会给学生上课准备零食。我想，最后一次课可能我会记很久，我们作为学生只是在做该做事情，老师却说我们学成这样把他感动到了。而且之后了解到教我们上机的学长都是南湖的，肖桐老师应该是来回接送，而且都是晚课，我想我更加明白了一些事情，我想，如果明年开课，我还是会继续去听的。

其次是感谢朱靖波老师，我想研究生的课堂里除了僵硬的知识 and 考试，应该给我们留下更多的思考，显然朱老师那一次的晚课分享，让我有了更多的思考，回去以后我想了朱老师讲的图像识别的实例，针对小孩子拍照乱动总是拍不好的问题，通过图片识别并打分，选出一系列照片中最符合美学的那一张。当然从效率上讲，无疑优于人工，但是对于小孩子而言，家长在意的更多的是乐趣，因此这个技术应用到更加专业的领域市场可能会更好，仅代表个人幼稚的想法。

最后感谢带我们上机的学长和我实验室的伙伴，一起坚持完了一个又一个困难的上机晚课，以及对我的帮助和指导，完成了一次又一次的上机实验，特别是那些我请教过很多次的学长，NMT 的翻译实例是张裕浩学长讲解的，第一次上机主要是单韦乔学长指导的，课下作业里胡驰学长提供了翻译实例上手的思路，第三次上机主要是马梦宇学长辅导的。最后感谢每次彼此陪伴坚守到最后的实验室小伙伴们！