把定点的代码，一层一层的替换为浮点的，看最后的结果

这种字准上升，句准下降看起来像是sil的先验概率不准，做实验之前可以先重点检查一下先验概率。

1）先抽取一些浮点代码中后面多字的情况的音频作为测试集

2）记录测试结果

3）先将定点代码中的定点化的sigmoid替换为float版本的sigmoid，看效果

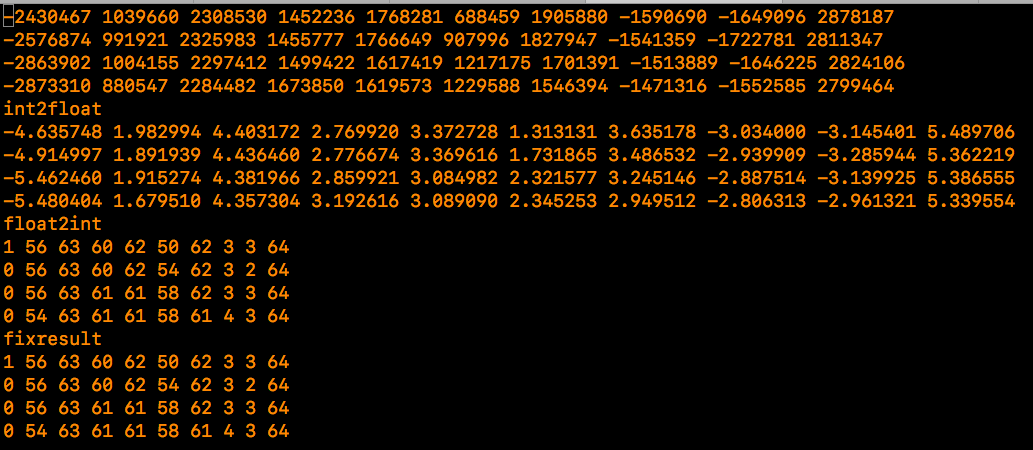
4）将定点代码中的wx+b的操作替换为float版本的wx+b，看效果

1. 看先验概率

首先可以确定先验概率相同。

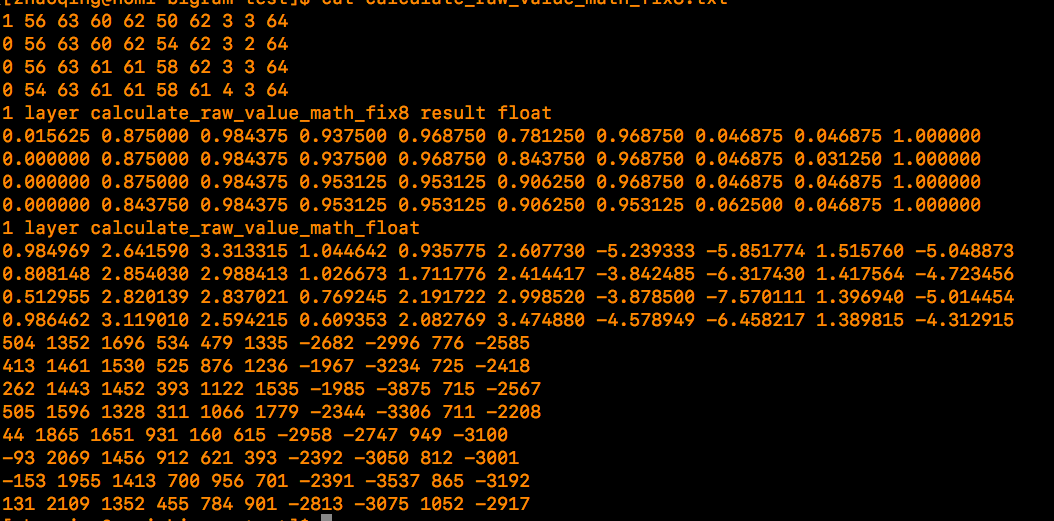
1. 把定点查表法的sigmoid与浮点计算法的sigmiod进行替换，对结果进行对比，结果一致。

vim src/feature/mlp/mlp\_neon\_matrix.h 239行

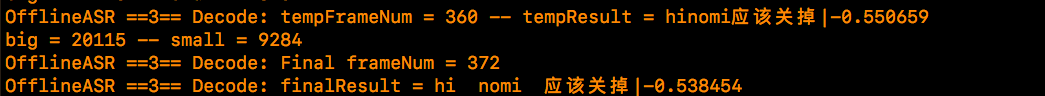


1. 定点代码中的wx+b的操作的替换为float版本的
2. calculate\_raw\_value\_math\_fix8替换为float的计算结果

新写一个函数来替换



原定点

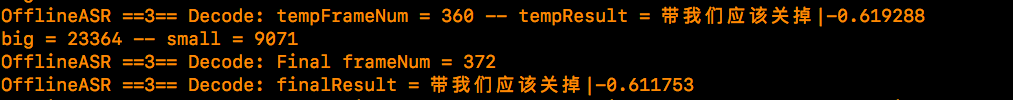


把所有的第二个定点的wx都替换为浮点后

图片包含 文字

描述已自动生成

浮点



单层替换，判断哪一层导致的定浮点差异较大。

见文件calculate\_raw\_value\_math\_fix8.txt为所打印的每一层的情况，每一层的输入均相同。

从输出结果可以看出，第一层与第二层在输入相同情况下，所得到的wx输出差距最大；可以定位到应该是1、2层w定点化后所造成的损失较大。

图片包含 墙壁, 文字

描述已自动生成

图片包含 墙壁, 文字

描述已自动生成

图片包含 文字, 墙壁, 记分牌

描述已自动生成

图片包含 墙壁, 文字

描述已自动生成

图片包含 文字, 墙壁

描述已自动生成

图片包含 文字, 墙壁

描述已自动生成

1. 上一步定位到定位到应该是1、2层w定点化后所造成的损失较大。

图片包含 电子产品

描述已自动生成

10 4 3 3 3 3 3 3

通过统计参数应该没有什么影响

Q为4，则有4位整数，3位小数，1个符号位，最小可以表示的非0小数为0.125

Q为3，则有3位整数，4位小数，1个符号位，最小可以表示的非0小数为0.0625

1\*pow(2,-3) = 0.125

1\*pow(2,-4) = 0.0625

定点为4的都加了一个0.5/（1<<3）= 0.0625

定点为3的都加了一个0.5/（1<<4）= 0.03125

0.043601 \* (1 << 4) + 0.5 #=1.197616

0.068608 \* (1 << 3) + 0.5 #=1.048864

在此计算之后，实际的可以表示的最小非0小数为

Q为4，0.0625

Q为3，0.03125

最大为

23.981834\* (1 << 3) + 0.5 = 192.354672

01111.111= 7.875

0111.1111= 3.9345

从参数大小上看，layer1与layer2定点化后为0的数较多，也就是表示不了的参数较多。

图片包含 文字

描述已自动生成

<0. 03125的数有140064+151075=291139

占比 291139/ 1048576 = 0.278

图片包含 文字

描述已自动生成

<0. 0625的数有33677+34347 + 62456=130480

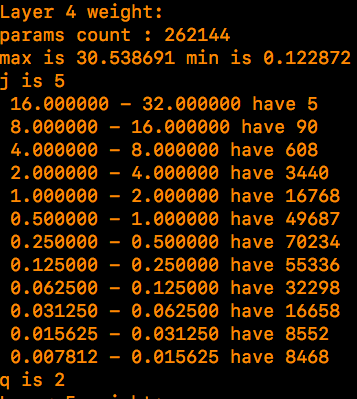
占比 130480/ 524288 = 0.249

图片包含 文字

描述已自动生成

<0. 0625的数有11344+11213 + 21937=44494

占比 44494/ 262144 = 0.17



<0. 0625的数有8552+8468+16658=33678

占比 33678/ 262144 = 0.128

图片包含 文字

描述已自动生成

<0. 0625的数有8295+8163+16324=32782

占比 32782/ 262144 = 0.125

图片包含 文字

描述已自动生成

<0. 0625的数有4505+4590+8814=17909

占比 17909/ 131072 = 0.137

发现浮点模型里的Q值与定点里面的不同

Float: 8 3 4 4 4 4 4 4

Fix: 图片包含 物体

描述已自动生成

修改float的定点化参数后，对比模型的信息

图片包含 室内

描述已自动生成

图片包含 文字

描述已自动生成

0.396478 \* (1 << 4) + 0.5 = 6.843648  
-0.152189 \* (1 << 4) - 0.5 = -2.935024

查了一下以前的数据发现浮点模型对应的模型是7300，最后确定定点和浮点的模型不是同一个。定点的为7500打包而成。

重新换为7500的浮点后再次进行比较，以上结果虽有意义，却是模型7300的定浮点结果。

现在先将浮点模型替换成7500后，跑测试中。

测试结果依然比定点的差，但比之前好了一些

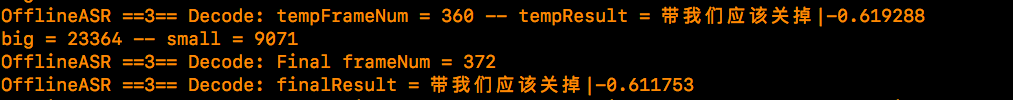
Lmscaling = 11



想起以前测效果的时候是以lmScaling = 13测试的，现在测全是用的11

3103178.wav 把氛围灯关掉

浮点

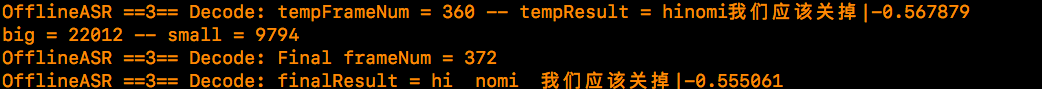


104333333

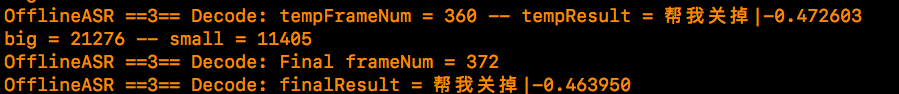
图片包含 火车

描述已自动生成

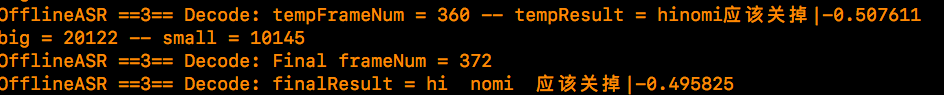
替换1



2



3



4

图片包含 监视器

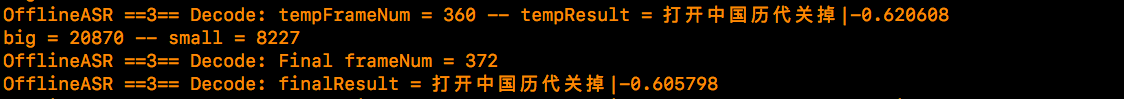
描述已自动生成

5

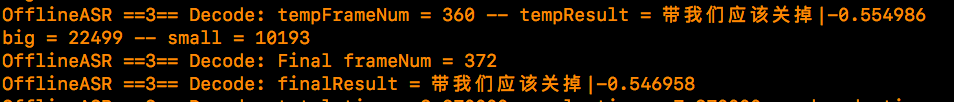
图片包含 火车

描述已自动生成

6



全替换



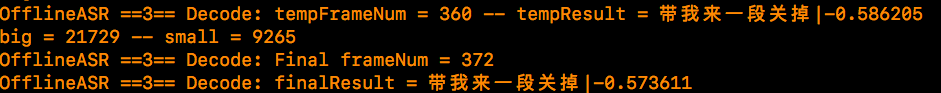
从结果可以看出，最大的不同出现在第4层，替换第4层后，定浮点的结果是相同的。

图片包含 文字

描述已自动生成

第4层的参数从统计上没有看出与其余层的不同，分布上肉眼看不出来。

调整第4层的定点化参数Q，结果如下



看起来差不多，重新跑一遍定点后的测试

vim src/feature/mlp/mlp\_decode\_cpu\_sse.cpp

vim src/include/res\_def/mlp\_decode\_define.h