# Bert服务使用说明

## 获得句向量

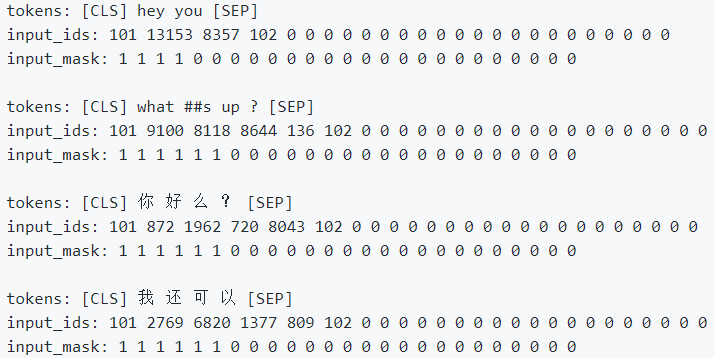
1. 启动客户端

from bert\_serving.client import BertClient  
bc = BertClient(ip="localhost")

ip表示服务端所在ip，由于是在本地运行，所以填localhost

1. 关于输入
2. .关于分词

Bert的中文官方预训练模型是以字符级别进行训练的，所以输入的中文语句不需要进行分词，即使有意地在两个单词之间加上空格，它也不会识别单词/短语。



1. 关于句子长度

句子最大长度可在服务端修改（默认为25，不能超过255），不足长度的句子将用0补齐（如上图），超过部分将被截断。

1. 池化层的策略（由服务端设置）
2. NONE

不会将所有的词向量进行压缩成句向量，当需要获得词/字向量的时候，选择此策略

1. REDUCE\_MEAN（默认策略）

该策略会对各个层的词/字向量的特征值取平均值

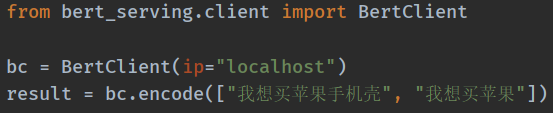
3）REDUCE\_MAX

该策略会对各个层的词/字向量的特征值取最大值

4）REDUCE\_MEAN\_MAX

分别以2）、3）两种策略获取特征值，将两个结果直接拼接成1536维度的特征。

1. 获得向量



句子的输入方式是list[“str1”,”str2”......]

在使用NONE策略的情况下：

返回的ndarray矩阵，规格是A×B×C（A代表输入的句子数量，B代表句子的最大长度，C表示特征数量（默认为768））：

vec[0] # [1, 25, 768],“我想买苹果手机壳”句子的句向量

vec[0][0] # [1, 1, 768], `[CLS]`标签的词向量（CLS是句子开始的标签，会被模型自动添加，针对下游任务对模型进行微调之后，此向量可以看做一个优秀的句向量）

vec[0][1] # [1, 1, 768], ‘我’的字向量

vec[0][2] # [1, 1, 768], ‘想’的字向量

vec[0][9] # [1, 1, 768], `[SEP]`标签的词向量（SEP是句子结束的标签，会被模型自动添加）

vec[0][10] # [1, 1, 768],填充标签’0’的字向量

vec[0][25] # 超出最大句子长度

在使用REDUCE\_MEAN（默认策略）的情况下：

vec[0] # [1,768],“我想买苹果手机壳”句子的句向量

vec[1] # [1,768],“我想买苹果”句子的句向量