

# 1 怎样情感分析

使用 Python 调用 SnowNLP 的库，我们就能用 SnowNLP 的功能来测试一句话的情感分数。

```
from snownlp import SnowNLP

s = SnowNLP("我喜欢 AI 技术")
print("分析得到的正向程度区间在 0~1 之间，0 是最负向，1 是最正向。")
print("这句话的正向程度：", s.sentiments)
```

运行结果：

分析得到的正向程度区间在 0~1 之间，0 是最负向，1 是最正向。  
这句话的正向程度：0.6127241594259307

```
s = SnowNLP("我觉得腾讯的技术最棒")
print("这句话的正向程度：", s.sentiments)
```

运行结果：

这句话的正向程度：0.9063186076606359

# 2 模型的假设和问题

关于今天使用的这个模型，我们选用贝叶斯分类器，它会根据词的频率和概率来判断感情色彩。但是这个模型的特定假设，它并不会关心到语言的顺序。我们举下面的例子。

```
s1 = SnowNLP("你骗了我")
print("你骗了我：", s1.sentiments)
s2 = SnowNLP("我骗了你")
print("我骗了你：", s2.sentiments)
print("'你骗了我' 和 '我骗了你' 情感一样？", s1.sentiments == s2.sentiments)
```

运行结果：

你不爱我：0.6105763619666702

我不爱你：0.6105763619666702

这两句话其实在情感表达上还是有区别的，但是因为这个模型忽略了语言的顺序，导致它的评判分数是一样的。