

## 1 简介

这次内容是一次实践内容，将会整合我们之前所有学的内容，包括数据处理，AI 技术，UI 交互技术等。具体的任务是制作一个收集男生女生不同情感的语句，然后让机器从积极到消极情感排序，展示这些富有感情色彩的语句，评一评男生女生中谁说的话是最积极的。



face

## 2 窗口设计

想想为了收集语句，展示语句，我们需要些什么窗口组件呢？

输入框，按钮，展示图片就能满足这次的需求了。

```
import tkinter as tk

window = tk.Tk()
window.title('情绪判断')
window.geometry('350x300')

e = tk.Entry(window, width=37)
e.grid(row=0, column=0)

b1 = tk.Button(window, text='分析', width=20, height=3)
b1.grid(row=1, column=0)

canvas = tk.Canvas(window, height=200, width=200)
image_file = tk.PhotoImage(file='./data/face_happy1.png')
image = canvas.create_image(100, 100, anchor='center',
                             image=image_file)
canvas.grid(row=2, column=0)

window.mainloop()
```

## 2.1 AI 计算感情强度

现在这个阶段我们按任何按键都是没有触发任何功能的。只有一个窗口程序的壳。为了实现 AI 技术，我们必须写出来这些承载 AI 技术的功能。

在点击“分析”按钮的时候，我们可以得到一个图片反馈，开心的情绪带来的是开心脸，伤心的是伤心脸等等，情绪的判断和高低程度都可以由你来控制。

在之前 AI 情感分析的课程中，我们介绍了使用 snownlp 来分析句子情感正向程度。现在就能实现语句输入+情感分析这个全流程了。

```
from snownlp import SnowNLP

def get_face_path(sentence):
    s = SnowNLP(sentence)
    score = s.sentiments
    print(score, sentence)
    if score < 0.25:
        path = "data/face_sad.png"
    elif score < 0.5:
        path = "data/face_unhappy.png"
    elif score < 0.75:
        path = "data/face_happy1.png"
    else:
        path = "data/face_happy2.png"
    return path
```

## 2.2 前端工程

现在我们就已经实现了这个程序的所有核心功能，接下来就是将这些核心功能衔接到窗口上去。有了上面的功能，我们再将这些功能插入到按钮中，就能实现按钮+输入框+情感标签输出的连贯功能了。

```
import tkinter as tk
from utils import load_extra_dict
from ai_face import get_face_path

load_extra_dict("./data/sentiment.marshal.3")

window = tk.Tk()
window.title('情绪判断')
window.geometry('350x300')

def get_face():
    global image_file
    sentence = e.get()
    path = get_face_path(sentence)      # 合并 AI 工程师的功能
    image_file = tk.PhotoImage(file=path)
    canvas.itemconfig(image, image=image_file)

e = tk.Entry(window, width=37)
e.grid(row=0, column=0)

b1 = tk.Button(window, text='分析', width=20, height=3,
command=get_face) # 注意这里会触发 get_face 的功能。
b1.grid(row=1, column=0)

canvas = tk.Canvas(window, height=200, width=200)
image_file = tk.PhotoImage(file="./data/face_happy1.png")
image = canvas.create_image(100, 100, anchor='center',
image=image_file)
canvas.grid(row=2, column=0)

window.mainloop()
```