

莫烦 python 的志愿者们

首先谢谢大家能够抽出自己的时间, 尽一份之力. 用实际行动支持开源, 造福更多的同学们. 其实写这个东西, 也相当于写一份自己的学习笔记清单. 自己回顾起来最方便. 我以前录视频的时候也是这么想的, 把录视频当做在线笔记, 忘了再回去看视频.

如果你们愿意, 我会在网页上打上你们的名字, 当做这一份笔记的 author. 我就当做小编啦~

下面是具体内容:

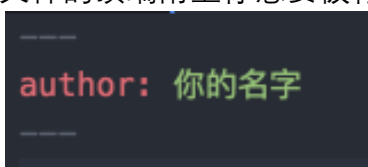
最终要出现的效果, 如[这个例子](#).

说白了就是视频的学习笔记, 用文字描述给大家这些代码是怎么用的. 大家不用管那些颜色特效是怎么出来的, 只需要按照特定的 markdown (.md) 格式写笔记, 我把你们写好的笔记 load 进我的代码, 特效就出来了.

笔记的模板文件叫做 markdown (.md), 可以直接用记事本打开. 如果想要预浏览你所编辑的内容, 你可以把 markdown (.md)文件放入 <http://dillinger.io/> 网站观看效果.

对于模板内容的解释:

1. 为了感谢你的支持, 你将会成为本 篇文字教程的作者, 我会在视频下面打上你的名字, 所以请在 markdown 文件的顶端附上你想要被称呼的名字.



2. 使用 ``some code`` 这样的格式会在文本描述中加上了 code 的效果, 用来强调这是从 code 里面来的.



3. 上者和代码高亮是不一样的. 上者只是在文字描述做变成代码高亮, 而真正的代码的高亮效果是这样实现的, 在代码区域的上面和下面加上 ````python` 和 `````. 注意在 ````python` 和 ````` 前后都需要空出一行, 起到隔离代码的效果, 不然代码就和上下

文黏在一起了. 而且代码每一行的跨度不要太长, 不然代码就溢出设定好的边框了.

```
python
# define placeholder for inputs to network
xs = tf.placeholder(tf.float32, [None, 1])
ys = tf.placeholder(tf.float32, [None, 1])
```

```
# define placeholder for inputs to network
xs = tf.placeholder(tf.float32, [None, 1])
ys = tf.placeholder(tf.float32, [None, 1])
```

4. 如果是这样排版, 在网上显示出来还是一段连续的句子, 中间不会空行.

接着我们将在这条曲线上面动手脚啦, 首先要定义一个方程 `def animate(i)` 来移动我们的 `line`. 移动时, 是基于变化的 `i`, 不同的 `i` 将会产生不同位置的 `line`.

接着我们将在这条曲线上面动手脚啦, 首先要定义一个方程 `def animate(i)` 来移动我们的 `line`. 移动时, 是基于变化的 `i`, 不同的 `i` 将会产生不同位置的 `line`.

不过如果是这样排版: 中间隔上一行. 他们就会变成一段一段的段落.

Tensorflow 是非常重视结构的, 我们得建立好了神经网络的结构, 才能将数字放进去, 运行这个结构.

这个例子简单的阐述了 tensorflow 当中如何用代码来运行我们搭建的结构.

首先, 我们这次需要加载 tensorflow 和 numpy 两个模块, 并且使用 numpy 来创建我们的数据.

Tensorflow 是非常重视结构的, 我们得建立好了神经网络的结构, 才能将数字放进去, 运行这个结构.

这个例子简单的阐述了 tensorflow 当中如何用代码来运行我们搭建的结构.

首先, 我们这次需要加载 tensorflow 和 numpy 两个模块, 并且使用 numpy 来创建我们的数据.

5. 在文中加链接的话是这样 [链接文字](链接网址)

看这个 [链接](http://matplotlib.sourceforge.net/api/animation_api.html)

具体信息请查看这个[链接](http://matplotlib.sourceforge.net/api/animation_api.html)

6. 如果你要用副标题, heading, 并出现这种效果:

什么是 Batch Normalization

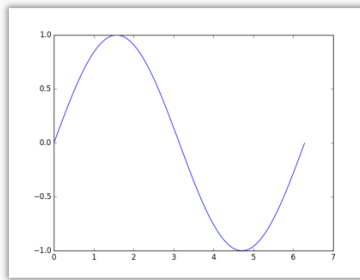
请参考我制作的 [Batch normalization 简介视频](#) Batch normaliz

我推荐大家使用 4 级标题, 也就是这样打:

“## 什么是 Batch Normalization”

7. 像 plt 中,我们会插入很多图片, 插图方法是这样: 以这样的形式 ``. 因为图片链接到以后还是要改动, 所以到时候就把文件和图片一起发给我, 你在 `src=`这个地方暂时打上图片的文件名, 我之后再改动.

```
[% highlight %]  
  
  
  
接着我们将在这条曲线上面动手脚啦, 首先要定义一个方程 `def animate(i)`
```



8. 更多关系 markdown 的介绍及使用规则,可以查看这里: <http://sspai.com/25137>
9. 如果你想提前预览一下自己编辑的 .md 文件效果, 请随便搜索“markdown online”,类似于这样的网站即可: <http://dillinger.io/>
10. 最后请你把你编辑好的文件(如果包含图片), 都一起打包给我~ 我在小小编辑一下,然后就 push 上莫烦 python 的网页啦~