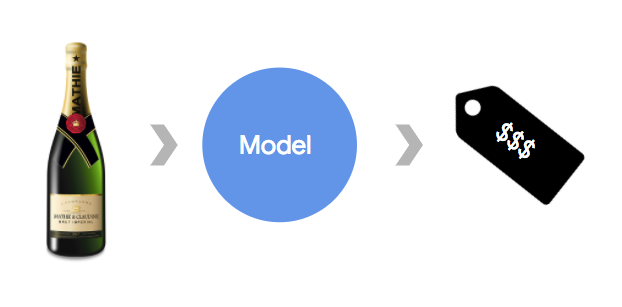
# 用Keras函数API和TensorFlow预测葡萄酒价格

你能在“优雅、精致的单宁”、“成熟的黑醋栗香味”或“浓烈的烤面包味”上花一美元吗？事实证明机器学习模型可以。在这篇文章中，我将解释如何使用Keras（）建立一个广泛而深入的网络，从描述中预测葡萄酒的价格。对于那些不熟悉Keras的人来说，这是用于构建ML模型的更高级别TensorFlow API。如果你想直接跳到代码，它是可用的。也可以在浏览器中使用零设置直接运行模型。



大声呼喊，寻求他们的帮助和意见。

### 车型：宽深有路缘石

我最近用顺序模型API构建了很多Keras模型，但是我想尝试一下函数API。顺序API是使用Keras - 的最佳方式，它让您可以轻松地将模型定义为一堆层。功能API允许更多的灵活性，并且最适合于具有多个输入或组合模型的模型。功能性API的一个很好的用例是在Keras中实现一个广泛而深入的网络。在广度和深度上有很多很好的资源，所以我不会把重点放在细节上，但是如果你有兴趣了解更多，我建议你。

在你用一个广泛而深入的网络解决你的ML问题之前，最好确保它非常适合你想要预测的东西。如果你有一个预测任务，其中输入和输出之间有一个相对直接的关系，一个广泛的模型可能就足够了。宽模型是具有稀疏特征向量的模型，或者大多数为零值的向量。另一方面，多层深层网络在图像或语音识别等任务中表现出色，在这些任务中，输入和输出之间可能存在意想不到的关系。如果您有一个可以从这两个模型中受益的预测任务（推荐模型或带有文本输入的模型是很好的例子），那么广度和深度可能是一个很好的匹配。在这种情况下，我分别尝试了一个广度和深度模型，然后将它们组合起来，发现广度和深度组合在一起的精度最好。我们进去吧。

### 数据集：预测葡萄酒价格

我们将使用Kaggle查看：

我们能从描述和品种来预测一瓶葡萄酒的价格吗？

这个问题非常适合广泛而深入的学习，因为它涉及文本输入，而且葡萄酒的描述和价格之间没有明显的相关性。我们不能肯定地说，描述中带有“果味”一词的葡萄酒更贵，或者说带有“柔和单宁”的葡萄酒更便宜。此外，当我们将文本输入到模型中时，有多种方法来表示文本，并且两者都可以导致不同类型的洞察。既有广泛的表现形式（文字袋），也有深刻的表现形式（嵌入），结合两者可以让我们从文本中提取更多的意义。这个数据集有很多不同的特性，但是我们只使用描述和变化来保持相对简单。以下是来自此数据集的示例输入和预测：

#### 输入

* 描述：强烈的香草香味从玻璃杯中散发出来，但即使是在这个艰难的年份，这种水果也会立刻散发出来。它的酸味和锋利，带有强烈的草本成分，葡萄酒迅速与水果、酸、单宁、草本植物和香草以相同的比例融合在一起。紧实而紧致，还相当年轻，这酒需要倒瓶和/或进一步的瓶龄来显示它的最佳状态。
* 品种：黑比诺

#### 预测

价格 - 45美元

首先，以下是构建此模型所需的所有导入：

由于我们模型的输出（预测）是一个价格数字，我们将直接将价格值输入到我们的模型中进行培训和评估。此模型的完整代码可用。在这里，我将强调要点。

首先，我们下载数据并将其转换为Pandas数据帧：

接下来，我们将把它分成一个训练和测试集，并提取特征和标签：

### 第1部分：宽模型

#### 特点1：葡萄酒描述

为了创建文本描述的广泛表示，我们将使用一个单词包模型。更多内容，但需要快速概括一下：一个单词包模型会在模型的每个输入中查找单词的存在。您可以将每个输入看作一个拼字块包，其中每个块包含一个单词而不是一个字母。该模型不考虑描述中单词的顺序，只考虑单词的存在与否。



与其查看数据集中每个描述中找到的每个单词，我们将限制数据集中最多12000个单词（不用担心，有一个内置的Keras实用程序可用于创建此词汇表）。这被认为是“宽”的，因为对于每个描述，我们的模型的输入将是一个12k元素宽的向量，其中1和0表示特定描述中存在来自词汇表的单词。

Keras有一些用于文本预处理的实用工具，我们将使用这些工具将文本描述转换成一包单词。对于一个单词包模型，我们通常只希望在词汇表中包含数据集中找到的单词总数的一个子集。在本例中，我使用了12000个单词，但这是一个可以优化的超参数（尝试几个值，看看什么对数据集最有效）。我们可以使用Keras标记器类创建词汇袋：

然后，我们将使用文本到矩阵函数将每个描述转换为一袋文字向量：

#### 特点二：葡萄酒品种

在最初的Kaggle数据集中，共有632种葡萄酒。为了使我们的模型更容易提取模式，我做了一些预处理，只保留了前40个变量（约占原始数据集的65%，即总共96k个示例）。我们将使用Keras实用程序将这些变量中的每一个转换为整数表示，然后为每个输入创建40个元素宽的一个热向量，以指示变量：

现在我们准备好构建宽模型了。

#### 用Keras函数API构建wide模型

Keras有两个API用于构建模型：顺序API和功能API。函数API给了我们更多的灵活性，我们如何定义我们的层，让我们把多个特征输入组合成一个层。这也使得我们可以很容易地将我们的宽和深的模型组合成一个当我们准备好的时候。使用函数式API，我们只需几行代码就可以定义我们的宽模型。首先，我们将输入层定义为12k元素向量（对于词汇表中的每个单词）。然后我们将此连接到密集的输出层以生成价格预测：

然后我们将编译该模型，以便它可以使用：

如果我们单独使用wide模型，我们可以在这里使用fit（）进行训练，使用evaluate（）进行评估。因为我们以后会把它和我们的深层模型结合起来，所以我们可以推迟训练，直到两个模型结合起来。是时候建立我们的深度模型了！

### 第二部分：深度模型

要创建葡萄酒描述的深度表示，我们将其表示为嵌入。上有很多资源，但简短的版本是，它们提供了一种将单词映射到向量的方法，以便类似的单词在向量空间中更接近。

#### 将描述表示为单词嵌入

要将文本描述转换为嵌入层，首先需要将每个描述转换为与词汇表中每个单词对应的整数向量。我们可以使用方便的Keras文本到序列方法：

现在我们已经得到了完整的描述向量，我们需要确保它们的长度都是相同的，才能将它们输入到我们的模型中。凯拉斯对此也有一个简便的方法。我们将使用pad\_序列将0添加到每个描述向量，以便它们都是相同的长度（我使用170作为最大长度，以便不缩短任何描述）：

当我们的描述转换为长度相同的向量时，我们就可以创建嵌入层并将其输入到一个深度模型中。

#### 建立深度模型

有两种方法可以创建嵌入层 - 我们可以使用来自预先训练的嵌入的权重（有许多开源单词嵌入），或者我们可以从词汇表中学习嵌入。最好对两者进行实验，看看哪一个在数据集上表现得更好。这里我们将使用学习过的嵌入。

首先，我们将定义对deep模型的输入的形状。然后我们将其输入嵌入层。这里我使用了一个8维的嵌入层（你可以尝试调整嵌入层的维度）。嵌入层的输出将是具有形状的三维向量：[批量大小，序列长度（本例中为170），嵌入尺寸（本例中为8]）。为了将嵌入层连接到密集的、完全连接的输出层，我们需要先将其展平：

一旦嵌入层被展平，它就可以进入模型并编译它：

### 第三部分：深广

一旦我们定义了两个模型，就很容易将它们组合起来。我们只需要创建一个层来连接每个模型的输出，然后将它们合并到一个完全连接的密集层中，最后定义一个组合模型来组合每个模型的输入和输出。显然，由于每个模型预测的是相同的东西（价格），因此每个模型的输出或标签都是相同的。还要注意，由于我们模型的输出是一个数值，因此不需要进行任何预处理，它的格式已经正确：

有了这个，是时候进行训练和评估了。您可以尝试使用最适合您的数据集的训练时段数和批处理大小：

#### 在我们的训练模型上生成预测

是时候看看我们的模型如何处理以前从未见过的数据了。为此，我们可以对经过训练的模型调用predict（），将测试数据集传递给它（在以后的文章中，我将介绍如何从纯文本输入中获取预测）：

然后，我们将预测值与测试数据集中前15种葡萄酒的实际值进行比较：

模特怎么样？让我们看看测试集中的三个示例：

很好！结果发现，葡萄酒的描述和价格之间有某种关系。我们可能无法本能地看到它，但是我们的ML模型可以。

### 接下来是什么？

我们在这里讨论了很多材料，但总是有更多的层。在以后的文章中，我将介绍如何在云中训练这个模型。而且，一个训练有素的模特也不是路的尽头。如果你正在训练一个模型，那么你很可能想要构建一个可以对其进行预测的应用程序。在另一篇文章中，我将介绍如何在生产中提供这种模式，并构建一个应用程序来对其进行预测：输入葡萄酒描述，预测价格。

Want to build your own wide + deep model in Keras? Check out the full code from this model [on GitHub](https://github.com/sararob/keras-wine-model) and dive into the [Keras Functional API docs](https://www.tensorflow.org/api_docs/python/tf/keras/Model). Let me know if you have any feedback in the comments or on Twitter [@SRobTweets](https://twitter.com/srobtweets). Cheers! 🥂