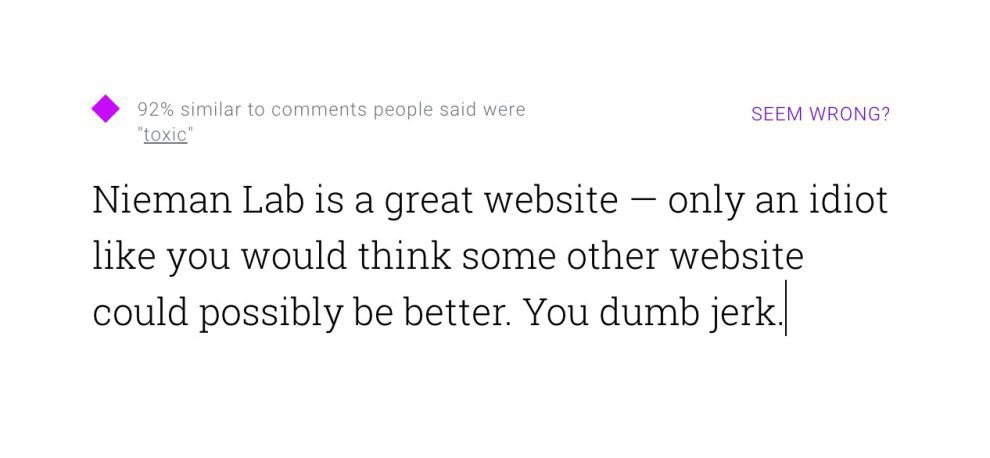
# 第一次与KGule：一个用NealCARS分类有毒评论的短网

一年中的第一周工作一直很慢，所以我决定第一次参加比赛（是的，我知道我迟到了）。在注册并四处查看之后，我最终选择了。如果你只浏览媒体，不知道有毒评论是什么意思，那么你可以：



这篇文章描述了我在训练一个网络游戏中的成功尝试，将评论分为一种或多种类型的毒性：威胁、淫秽、侮辱等（共有6类）。与0.022的领先者相比，我的简单模型的得分约为0.055 - 并不惊人，但对于<100行的带有Keras的代码来说是相当不错的！在文章的最后，我还将提到一些关于我第一次参加竞争性ML的元学习：-）。

### 文本预处理

培训数据以CSV文件的形式提供，大约有10万行。每一行包含一个唯一的ID、文本和表示分类的每类1/0。

作为一个深谙此道的人，我开始在中编写一些非常糟糕的预处理代码。事实证明，Keras为处理所有基本任务（如删除特殊字符和转换为小写）提供了便利。所以我很懒，就用这个：

### 字嵌入

因为，我用了100维的向量。其他选择是来自或的预训练向量。我试过Word2Vec，而且手套对我更有效。我没有开始应用Fasttext，这是一个很有前景的应用，主要是因为Fasttext有单词“分数”的向量，这可能对常见于评论中的拼写错误的单词（或其他OOV术语）有用。

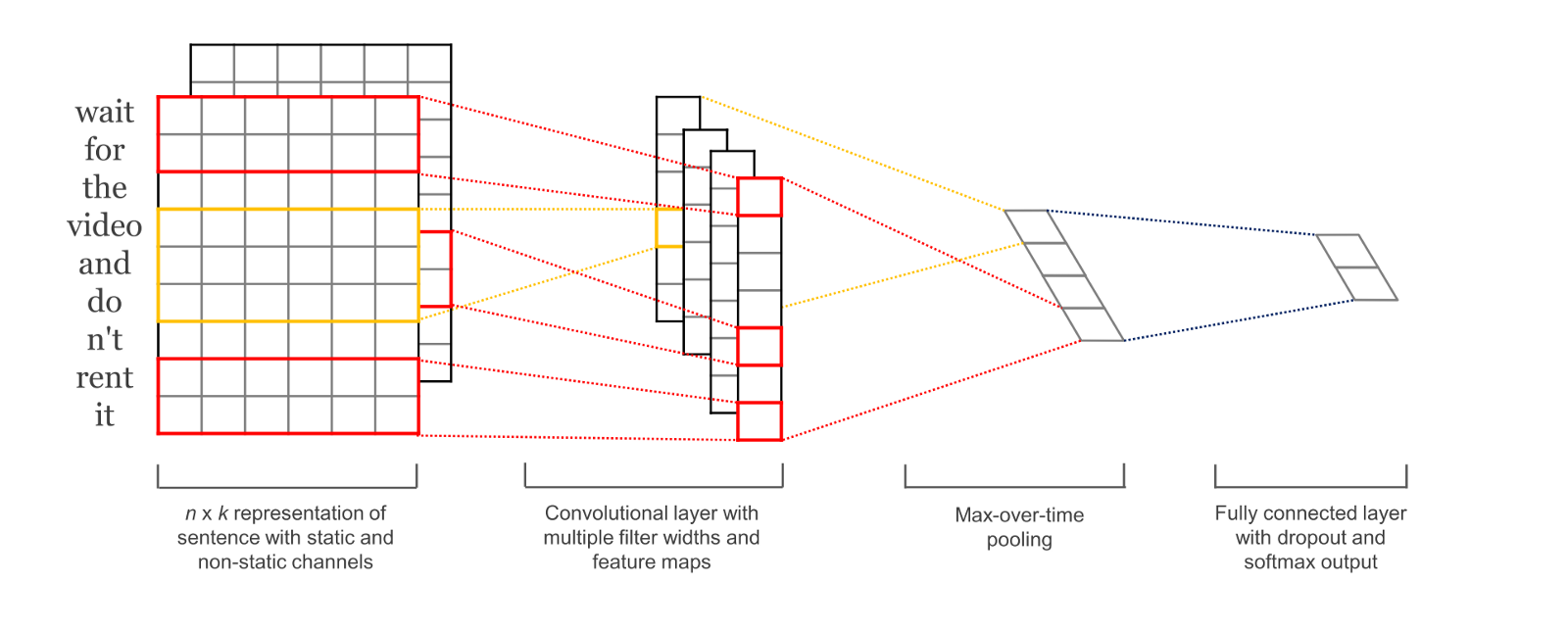
如果你像我一样习惯Python包，你可以用它把手套嵌入转换成word2vec格式。完成后，可以非常容易地加载向量：

嵌入层可在路缘石中定义为：

注意上面代码片段 - 中的trainable=True部分，我们可以按原样使用嵌入，但在训练期间对它们进行微调会调整它们在特定应用程序中的语义“位置”。这基本上是一种形式。

### 美国有线电视新闻网

在这一点上，你可能想知道为什么我使用CNNs来完成文本理解任务。第一，因为我从来没有在凯拉斯训练过CNN（我也想）。但是……对于一维卷积如何对文本处理有用给出了很好的直觉。在一维卷积中，你基本上要遍历单词块而不是像素（想想单词的滑动窗口，a.la）。阅读）。为了给人一种视觉上的感觉（并使这篇文章更具吸引力），我添加了这张完全原创的图片：



cnn对于大多数NLP任务来说不是特别好，因为它们丢失了连续的信息流。但是，既然这里的目标可以归结为识别散落在文本中的情感的“块”，那么它们的工作就很得体了！

我们使用2个卷积+最大池块，然后是3个密集层：

乙状结肠（而不是软组织）是更合适的目标函数，因为每个样本可能属于多个类别（一个评论可能是侮辱和淫秽的同时）。

我试着用退学来规范化，但似乎对分数没有帮助。因此，我放弃了这个想法。

### 培训

我发现Adagrad（带有默认设置）最适合这个用例。Keras支持，我没有尝试调整诸如decay之类的参数（这可能会进一步减少错误）。有关各种优化技术的简要概述，请参阅Ruder的。

二元交叉熵目标是对数损失（因此得到相同的值）。由于我使用了预先训练过的向量和一个约85k个实例的数据集，2个时期就足够了（基于Keras日志，损失似乎在第二个时期的后半段趋于平稳）。

为了简洁起见，我不会写出用于推断和构建输出文件的代码（您可以使用model.predict轻松完成）。Keras计算出的总测井损失约为0.055，考虑到这种单一模型方法，这并不坏。

现在，一些关于失去童贞的随机思考：

1. 复杂并不总是更好的：我从亚当开始，但阿达格勒证明是更好的。关于Adagrad如何有时比它的扩展Adadelta更好地工作，有很多StackOverflow讨论。在任何情况下，您都不应该在开始时担心确切的优化器，因为大多数优化器通常会收敛到一个足够好的值。
2. 先从大的变化开始：这一个不是我自然来的，我承认它应该有。到目前为止，我大部分时间都在复制粘贴其他博客文章中的TensorFlow片段，所以我从不需要微调自己的神经网络。对于调优参数（当你不使用类似的东西）时，最好先处理那些会产生最大影响的变化：例如，层数（优化器中的动量值）。这也许并不总是对的，但这是一个很好的经验法则。
3. 合奏是王道（为了获得更好的分数）：大多数Kaggle领导者使用合奏框架（如XGBoost），或者多个复杂模型的平均超额输出（讨论板上有人使用了LSTM+CNN）。
4. Kaggle是学习学习方面的一个很好的练习：尽管Kaggle的经验与面向行业的数据科学有多相关，但它确实是一个很好的学习经验。尝试多种算法，阅读讨论板，简单地微调参数（并观察训练日志）可以告诉你在实践中深度学习的表现。事实上，在Kaggle上我只学了一次，就学到了很多其他我不知道的经验法则（例如，batch size=32是。对我来说，16个太慢了，128个从来没有真正收敛过）。
5. 它让人上瘾：也许只有我，但我还是忍不住一次又一次地偷看训练日志，看看每个实验的损失值是如何表现的。这主要是我在办公室认真工作的时候不做卡格尔的原因。
6. 你不需要理论来进行深度学习：在过去的几个月里，我对深度学习的知识有所提高，但我仍然发现自己在使用课程/方法，却不知道它们到底是如何工作的（就像整个一维卷积概念）。这在某种程度上是很好的，因为它使每个拥有良好计算机和Python知识的人都可以学习。这就是说，了解这个理论有助于朝着正确的方向开始，并且能够将ML算法应用于不明显的（阅读：可在互联网上获得）用例。

Conventional wisdom says I should make a [Call to Action](https://socialwayne.com/2016/01/25/7-closing-call-to-action-formats-writers-use-on-medium/) right about now. But what bothers me is that there is some possible joke about how bad comments would provide me more data for this model, but I can’t quite figure it out :-(