

# Ai 训练营-nlp项目内容

## Week2：手写作文自动评分

- 学习使用textblob
- 学习readability的计算方法和相关工具
- 自动作文评分理论与方法学习

## 学习指引

### Day 1: 安装并学习使用 textblob 工具，学习parsing相关知识

今天我们将学习使用另一个nlp常用工具textblob，通过使用这个工具，学习其中的Part-of-speech Tagging，Spelling Correction，n-gram等概念

1. 阅读nltk book 第八章节，学习parsing概念

<http://www.nltk.org/book/ch08-extras.html>

2. 安装textblob

<http://textblob.readthedocs.io/en/dev/install.html>

3. textblob功能点学习

<http://textblob.readthedocs.io/en/dev/quickstart.html#quickstart>

### Day2: 学习Readability的定义和计算方法，学习使用工具textstat 和 readability。

今天我们学习nlp中文章可读性的定义和计算方法，同时会用工具去计算。文章的可读性在自动作文评分，分级阅读等领域都有应用。

1. 学习readability的定义

[https://en.wikipedia.org/wiki/Readability\\_test](https://en.wikipedia.org/wiki/Readability_test)

2. 学习使用工具 textstat，

<https://pypi.python.org/pypi/textstat/>

3. 学习使用工具 readability

<https://pypi.python.org/pypi/readability>

### Day3: 学习Keras的安装和使用

今天我们学习Keras,掌握一种快速搭建和使用深度神经网络的工具，可选择用于后面建立基于深度学习的自动作文评分

## 1. Keras 的介绍和安装

中文：<http://keras-cn.readthedocs.io/en/latest/>

英文：<https://keras.io/#installation>

## 2. 学习使用keras

[http://keras-cn.readthedocs.io/en/latest/getting\\_started/sequential\\_model/](http://keras-cn.readthedocs.io/en/latest/getting_started/sequential_model/)

## 3. 使用keras对文本进行处理和编码

<https://keras-cn.readthedocs.io/en/latest/preprocessing/text/>

# Day4&Day5:阅读论文，学习自动作文评分的理论和方法

前面几天我们已经学习了可以用来进行自动作文评分的一些概念和方法，接下来，我们通过阅读几片论文，了解和学习目前自动作文评分的主流的方法和理论。人工进行特征选择的方案，能够给写作文的用户带来更多的反馈信息，一定程度可以指导用户在弱项上进行提升；相比之下，基于深度学习的方法可能可以得到好的性能，但是是一个黑盒，从中没有办法获得更多的反馈；大家根据个人情况，可以充分发挥，例如做一些混合的系统等。以下内容作为同学们实现系统的参考：

参考阅读材料：

人工进行特征选择的评分系统：

[https://www.researchgate.net/publication/278383803\\_Task-Independent\\_Features\\_for\\_Automated\\_Essay\\_Grading](https://www.researchgate.net/publication/278383803_Task-Independent_Features_for_Automated_Essay_Grading)

基于深度学习方法的评分系统：

[http://www.aclweb.org/old\\_anthology/D/D16/D16-1193.pdf](http://www.aclweb.org/old_anthology/D/D16/D16-1193.pdf)

目前市面上比较权威的一个系统：ETS的e-rater评分系统：

<https://www.ets.org/Media/Research/pdf/RR-04-45.pdf>

# Day6: 设计方案，进行机器自动评分

下面的链接是使用基于深度学习的方法进行作文自动评分的样例（前面提供论文的一个实现），可以作为一个实现方案的参考，同时大家可以自行设计自己的网络拓扑去做一些实验尝试。

此外，需要注意的是，我们选择的数据比样例少，所以考虑使用cpu模式去做实验，而样例使用了gpu模式。

参考链接：

<https://github.com/nusnlp/nea>