人工智能课堂总结

杨昆霖 2015201914

# 课程收获

这门课主要介绍了机器学习与算法与一些深度学习算法的基础知识，课下使用Scikit-Learn与TensorFlow进行算法运用，同时以AI Car作为课程项目，极大锻炼了动手能力，将理论与实践相结合，算是本科期间收获最大，实用性最强的一门课之一。

课堂带给我的收获主要分两方面，一方面是在已有的对数据挖掘算法的再更深了解，另一方面是深度学习算法的初步了解，下面将分别阐述。

这门课之前，我对机器学习算法的主要应用是利用Weka工具包跑一些简单的模型，对人工智能领域的详细研究步骤并不了解。通过这门课，我知道事实上人工智能的研究也是有迹可循的。首先是要对所研究的问题进行框架化与细节化，不能只追求一个结果的准确性，事实上结果正确与否并不是唯一的衡量标准，要先弄清楚这个模型的运用之处以及最终的产品要如何从该模型中受益。同时在盲目套算法之前，要先研究清楚当前解决该问题的方法，如果有的话，再在此基础上进行一定的优化操作，比如将传统的基于规则的算法改成机器学习算法等。

同时，根据问题的不同，衡量方式也差异很大，有的可能是准确度，有的可能更注重召回率，有的或许注重性能与增量学习等方面，先确定衡量标准之后，才能进一步地开始研究。再次，考察数据之前，要对解决的问题进行一个假设和验证，这样在真实情况下碰到数据才不会一头雾水。虽然现实项目碰到的数据集可能不那么规整，但一些像UCI、ImageNet这样的数据平台就有很多规整化的数据。

得到数据集之后，要先进行数据清洗，并划分测试集与训练集，有的时候还需要对数据进行规范化处理。这些工具在Scikit包中都有着对应的接口。由于人工智能是用于处理大量数据获得精准预测模型的，所以数据集往往维度较高，单纯肉眼无法分辨出其中的数据分布等信息，更不知道如何对数据进行处理。这时就需要利用数据可视化工具与相关性分析，对数据的分布与两两之间的关系进行一个初步分析，甚至是将属性组合起来分析，这样就可以对数据有一个更为直观的了解。通过这学期的实践，我现在已经能基本熟练使用matplotlib进行跑模型前的可视化分析操作。

这一切准备工作做好之后，才算是开始真正的机器学习模型构建阶段，因为之前的步骤与社科等领域的研究方法基本类似。由于机器学习模型都是基于数值的，一方面会对噪声数据与较大的数据较为敏感，另一方面对于类别型数据也较不好处理。这样就需要进行数据的清洗与转换操作。其中数值型的数据要进行规范化操作，有些变量可能还需要取log后再进行映射处理，因为可能对于变量我们更关心相对大小而不是绝对大小。对于类别型数据，就要采取哑变量变换操作，或者对于文本可能就要将其转为词向量等。特征转换完成之后，还需要进行特征归一化操作，这样避免数据中极端值的影响。这学期有次在实现BP神经网络算法的时候就遇到过这个问题，特征提取好之后利用神经网络进行训练却发现梯度怎么也不下降，后来发现原来是因为特征没有归一化，有些值特别大的特征影响了模型整体的训练效果，使得其他值相对较小的特征几乎没变化甚至变得更加趋近于零。这足以体现数据归一化的重要性。

对于模型训练方法，也需要下点功夫。模型构建这里，有多种训练测试法，常见的是多折交叉检验等，这些在Scikit包中也有现成工具，不过我线下也自己实现了一遍，仿造scikit的CrossValidation源码，一方面通过实践感受到这种数据划分法的优势，另一方面也能体会到别人代码的优越性，和自己写的还是差别挺大的。不过个人认为，对于预测性数据，比如根据用户过去交易预测未来交易的数据，可能不大适合这种交叉检验方法，因为逻辑上来说，要先知道过去才能预测未来，而不是根据未来再返回来预测过去。当然对于普通的分类问题，交叉检验是有着很大的优越性的。

前面这些模型训练的基础完成之后，就是模型训练的重头戏了。这学期学的主要普通机器学习算法模型有Linear Regression、Lasso Regression、Polynomial Regression、Logistic Regression、Support Vector Machine、Decision Tree和Random Forest等方法。就个人学习状况而言，实践过的算法有Linear Regression、Logistic Regression、Support Vector Machine、Naïve Bayes、Decision Tree和Random Forest，其中手动实现过C4.5 Decision Tree。通过使用这些算法，直观的感受是线性算法的性能依赖于算法的复杂度，集成学习普遍由于普通学习算法，基于概率的算法较为稳定。

具体而言，线性模型中，效果通常是SVM>Logistic Regression>Linear Regression，复杂度也是依次递减的。虽然SVM等算法在现在的机器学习研究论文中通常作为baseline来比较，但其经典与成熟的算法模式使得其在现在依然有着广泛应用，也不断有人在对其进行改进，如最近了解到的一个one-class SVM，用于进行异常点检测的，应付训练样本中负样本及其少的情况。同时SVM也可以利用核函数改成非线性模型，还能用来进行回归操作。

对于决策树算法，我有对其进行实现过，所以印象比较深。该算法有一个很大的特点是对其的特征预处理操作可以较少，尤其是数值化特征的归一化操纵，因为它是树状结构的决策分类，每次只判断一个数与判定条件的大小关系对其分类，因此不需要哑变量处理或者归一化，就算进行了那些操作对其效果影响也不大。

决策树的一个典型的扩展就是随机森林，也是现在效果又好训练又快的一种算法，在有些数据集上，随机森林效果甚至于深度学习算法差距不大。随机森林的主要思想就是利用了集成学习的“少数服从多数”机制以及其在进行单独每颗树学习时的特征丢弃操作。这些至少带来如下两点好处：一方面是集成学习方法使得学习器的性能准确率通常高于单一学习器，这就好比同一道难题，让一个专家来评判与让多个不同领域的专家来评判最后进行举手表决一样，显然后者优于前者。另一方面是随机森林算法在每颗树的训练当中随机的去除一些特征，这样不仅减少了模型的过拟合程度，还使得每颗树模型学到的特征差异较大，从而增加了最后做决策的准确率。

机器学习算法之外，还需要对特征进行降维处理。其实降维不仅仅能加快模型的训练速度，还可以一定程度上增加模型的准确率。这可以这么理解，维度高会导致其中的一些不相关的特征与相关性强的特征具有同样的地位，容易使得模型过拟合，适当降维就可以去除这些不必要的影响因素，反而可以增加模型的准确率。就好比阅读一篇英文的长难句，如果直接分析冗长的从句与定语的话，很难一针见血地看出表达意图，但是如果将其缩写的话就一目了然了。常见的降维方法有LLE和PVA。关于降维我印象较为深刻的是在推荐系统中的矩阵SVD分解，这算是降维的一种很直观的运用。

机器学习算法之后，便是深度学习算法，由于此前运用较少，现在也在实践当中，体会就没那么深刻。我对深度学习的理解是，深度学习是神经网络算法的极大提升，因为其引入了神经元这个概念，因此就使得深度学习的算法种类以及模型较为多样。神经元之间可以相互组合传递信息，其组合方式可以是自上而下的，可以是团簇型的，也可以是链条状的，由此便产生了深度神经网络、卷积神经网络与循环神经网络。深度神经网络与普通的神经网络结构最为接近，神经元自上而下一层层得连接，层次之间相互传递信息，层次之内不传递信息，算是传统分类问题的提升。卷积神经网络利用卷积方式使得神经元以矩阵团簇的形式连接，神经元与其周围的神经元发生信息传递，一层层得卷积上去，并且有池化操作等，通常用于图像识别等各个特征与周围邻近特征关系紧密的分类。循环神经网络则是神经元排成一列，神经元之间相互可以进行信息传递，可以看做神经元之间有着短路。又由于信号传递方向固定，前面的神经元有可能将信号传递给后面的神经元，只要对应的“门”没有关闭。

由于深度学习的层数与特征的增多，训练的要求远远大于普通的机器学习算法，听说完整的中文语音识别模型可能需要整个集群训练上千个小时才行。但其在图片、音频与自然语言处理方面的强大显而易见。由于课堂教到深度学习模型的时候已经临近期末，自身锻炼不大充足，因此打算等寒假再细细研究。现在利用课堂的知识，已经初步开始对图像和音频搭建简单的深度学习模型，进行识别分类操作。

# 课程项目

在这次的课程项目中，我是我们组的组长。对于我们的课程项目设计，我主要是想在完成老师的基础要求之上，每次再增加至少一个扩展功能。

## Random Walk AI Car

这次任务主要是将小车拼装完毕，并且能够利用蓝牙模块用手机和电脑操作小车，还能使用超声波传感器进行随机游走。我们在完成了基础功能之上，我又添加了基于超声波传感器的自动走迷宫功能和基于超声波传感器的区分运动与静止物体的功能。

**基础功能**

| **功能** | **实现效果** |
| --- | --- |
| 拼接组装 | 小车配有左右两个带马达的前进轮与一个导向轮，供电采用四节5号电池，利用Arduino模块附有超声波模块与蓝牙模块 |
| 超声波传感器 | 利用声速与时间的关系，计算出与前方物体的距离 |
| 蓝牙模块 | 可以利用电脑或者Android手机对小车进行相应操控 |
| 随机游走 | 能在地上随机行走，碰到障碍物转弯或者掉头，保证全程不碰到障碍物 |

**附加功能**

| **功能** | **实现效果** |
| --- | --- |
| 自动走迷宫 | 利用障碍物摆一个迷宫，有一个入口和一个出口，只要道路够宽敞，小车可以在较短的时间内从一个口进入，另一个口出来，期间不碰到迷宫的边缘障碍物 |
| 识别运动物体 | 随机游走时，判断前方物体是障碍物还是运动物体，如果是障碍物就绕开随机走，如果是运动物体就加速冲上去直到撞击为止，之后再继续随机游走 |

## Sensored AI Car

这次是要利用Scikit-Learn工具来为小车提供机器学习算法，使得小车可以利用简单的机器学习算法进行导航。我们使用了OpenCV加上Scikit包来让小车拥有“视觉”。我们利用一只宠物鼠为道具，用小车识别出老鼠，并对其进行追踪。

我们实现的功能如下：

| **功能** | **实现效果** |
| --- | --- |
| 实时影像传输 | 使用DroidCam将手机摄像头画面传输到网页，再使用爬虫爬取画面储存到本地 |
| 蓝牙指令发送 | 使用python包pybluez操控电脑通过蓝牙发送指令 |
| 运动小鼠追踪 | 使用hog+svm图像识别算法识别出画面中的小鼠 |
| 运动轨迹标注 | 根据拍摄到的视频，自动判断出运动的物体并对其进行相应的像素标注与运动方向的展示 |

## Deep AI Car

这次主要需要利用TensorFlow包来进行深度学习，为小车加上更强的感应功能。这里我们使用了OpenCV来处理图像，librosa来处理音频，使用TensorFlow来对处理结果进行深度学习。我们希望小车能够进行室内与室外导航功能，在室外沿着跑道行走，在室内能够利用摄像头自动躲避障碍。

我们实现的功能如下：

| **功能** | **实现效果** |
| --- | --- |
| 识别语言 | 使用者喊“Hey AI”，AI Car能够识别该话语并作出对应反应 |
| 识别使用者 | 利用第二次试验中所做的运动识别算法，识别出感兴趣的运动点，并利用机器学习算法识别出这些点的特征，使得之后可以跟踪使用者行走。 |
| 室外导航 | 使用TensorFlow搭建卷积神经网络，识别出跑道以及跑道的延展方向，从而使得车可以沿着跑道行走。 |
| 室内导航 | 利用两个摄像头，实现双目测距功能，在遇到障碍物时能自动做出反应。 |