****

本科毕业设计（论文）

**基于深度学习的足球运动员比赛表现综合评分模型研究**

|  |  |
| --- | --- |
| **学院**  **专业**  **学生姓名**  **学生学号**  **指导教师**  **提交日期** | **数学学院** |
| **信息管理与信息系统** |
| **蒙安哲** |
| **201530431306** |
| **凌黎明** |
| **2019年6月3日** |

摘 要

众所周知，足球是第一运动，也是世界上最流行的体育项目。正因如此，合理有效地综合衡量一位足球运动员的表现水平才会受到来自经济、竞技甚至学术层面上的关注。另一方面，基于知识发现这一目的的机器学习甚至深度学习等方面的研究随着当今数据体量日益庞大、数据挖掘需求日益显著而备受关注。目前，随着足球场上数据采集技术日益进步，海量的足球比赛相关信息的产生为机器学习、深度学习被引入足球领域创造了条件。本次研究主要工作包括根据通过网络爬虫所获取的中国足球超级联赛过往的赛事数据，以及分别应用较为普遍的卷积神经网络和长短期记忆循环循环神经网络构建足球运动员综合评分模型。

研究过程将按照以下步骤进行：选择数据源、搭建Scrapy网络爬虫框架、数据采集、数据预处理、CNN建模以及LSTM建模。其中，数据预处理工作重点在于采用主成分分析法对数据进行降维。本文以均方根误差和预测偏差的标准差作为评价指标，比较分别基于CNN与LSTM建立的评分模型的优劣程度。另外，在CNN模型与LSTM模型分别建模的过程中，本文还会介绍到如何通过选择合适的激活函数来提升模型效果的优化方法。

本研究成功建立的球员综合评分预测模型有助于球队教练、球队总经理等足球从业者根据较容易获取的足球赛事基础信息来获知球员的综合水平，进而在战术部署、球员转会操作甚至赛况预测等方面都大有用处。

关键词：足球；网络爬虫；主成分分析；卷积神经网络；LSTM

Abstract

It is widely acknowledged that football, or soccer to North American people, is the biggest sport as well as the most popular one in the world. Therefore, reasonably measuring the performance of a player on pitch is economically, athletically and even academically interesting. What is more, with the data constantly emerging as well as the demand of data mining in today’s world, machine learning and deep learning which aims at knowledge discovery is increasingly focused. Meanwhile, the huge-volume data related to the games enables the introduction of machine learning and deep learning to the field of football, given that data-collecting techniques are more and more mature. As for my study, there will be mainly two procedures: acquring the data of China Super Football League from a certain website based on the web crawling techniques, and respectively applying Convolutional Neural Network (CNN) and Long Short-Term Memory Recurrent Neural Network (LSTM), which are both widely utilized algorithms among the field of deep learning, to modeling the comprehensive rating of a player’s general performance.

Throughout the study, the process will go through from the selection of data source, building Scrapy web-crawling framework, pre-processing, CNN-based modeling to LSTM-based modeling. Among them, the key point of pre-porocessing of data is the reduction of the dimensions of the dataset with the method of Principal Component Analysis (PCA). Later by comparing the goodness of fitting between CNN-based model and LSTM-based Method, I am going to judge the quality of the model. It is noteworthy that the measure that I am going to judge with is Root Mean Square Error (RMSE). In addition, while building the CNN-trained and LSTM-trained models respectively, there is an optimization also going to be introduced, which basically selects the most suitable activation function in order to improve the quality of the models.

It is hopeful that either of the models of player’s comprehensive rating built throughout the study is helpful to those practitioners in the field of football including coaches and team mangers, due to the fact that the models provide them with an intuitive judgment of a player’s performance level simply according to the basic data which is easily available. As a result, the model is therefore meaningful in terms of data-supported tactic deployment, transfer operation and even match prediction.

**Key words:** football; web crawling; PCA; CNN; LSTM

目 录

[摘 要 I](#_Toc9929252)

[Abstract II](#_Toc9929253)

[第一章 绪论 1](#_Toc9929254)

[1.1 课题研究背景及其意义 1](#_Toc9929255)

[1.2 研究现状 3](#_Toc9929256)

[1.2.1 关于综合评分的研究现状 3](#_Toc9929257)

[1.2.2 关于深度学习在体育领域的研究现状 3](#_Toc9929258)

[1.2.3 关于深度学习在足球领域的研究现状 4](#_Toc9929259)

[1.3 本文研究内容以及主要贡献 5](#_Toc9929260)

[1.4 文章主要内容以及章节安排 6](#_Toc9929261)

[第二章 有关理论基础 8](#_Toc9929262)

[2.1 网‌络‌爬‌虫 8](#_Toc9929263)

[2.1.1 网‌络‌爬虫‌原理简‌介 8](#_Toc9929264)

[2.1.2 网络爬虫的相关技术 9](#_Toc9929265)

[2.2 主成分分析 11](#_Toc9929266)

[2.2.1 主成分分析的数学模型 11](#_Toc9929267)

[2.2.2 因子分析与因子旋转 13](#_Toc9929268)

[2.3 卷积神经网络 14](#_Toc9929269)

[2.3.1 CNN的拓扑结构 14](#_Toc9929270)

[2.4 LSTM 16](#_Toc9929271)

[2.5 激活函数 17](#_Toc9929272)

[2.5.1 RELU 17](#_Toc9929273)

[2.5.2 Softplus 18](#_Toc9929274)

[2.5.3 ELU 19](#_Toc9929275)

[2.5.4 Sigmoid函数 20](#_Toc9929276)

[2.5.5 Tanh函数 20](#_Toc9929277)

[2.5.6 Softsign函数 21](#_Toc9929278)

[2.6 小结 22](#_Toc9929279)

[第三章 数据的获取与处理 24](#_Toc9929280)

[3.1 数据源的选择 24](#_Toc9929281)

[3.2 Scrapy爬虫项目设计 24](#_Toc9929282)

[3.2.1 搭建Scrapy工程 24](#_Toc9929283)

[3.2.2 信息爬取的具体实现 26](#_Toc9929284)

[3.3 数据预处理 29](#_Toc9929285)

[3.4 缺失值检测 30](#_Toc9929286)

[3.4.1 重复值检测 31](#_Toc9929287)

[3.4.2 数据形式变换 32](#_Toc9929288)

[3.4.3 数据降维 32](#_Toc9929289)

[3.5 小结 35](#_Toc9929290)

[第四章 基于卷积神经网络的评分模型 36](#_Toc9929291)

[4.1 实验准备 36](#_Toc9929292)

[4.1.1 软件依赖 36](#_Toc9929293)

[4.1.2 评价标准 36](#_Toc9929294)

[4.2 训练集与测试集数据的准备 37](#_Toc9929295)

[4.3 卷积神经网络的构建 38](#_Toc9929296)

[4.4 卷积神经网络模型的优化 39](#_Toc9929297)

[4.5 实验结果与分析 40](#_Toc9929298)

[4.6 小结 41](#_Toc9929299)

[第五章 基于LSTM的评分模型 43](#_Toc9929300)

[5.1 实验环境 43](#_Toc9929301)

[5.2 LSTM模型的构建 43](#_Toc9929302)

[5.3 LSTM模型优化实验结果与分析 45](#_Toc9929303)

[5.4 小结 47](#_Toc9929304)

[第六章 总结与展望 48](#_Toc9929305)

[6.1 本文主要工作与结论 48](#_Toc9929306)

[6.2 实验中存在的不足与对未来研究的展望 49](#_Toc9929307)

[参考文献 51](#_Toc9929308)

[致谢 54](#_Toc9929309)

# 绪论

## 课题研究背景及其意义

众所周知，足球是第一运动，也是世界上最流行的体育项目。而在林林总总各类赛事中，最受人关注的当属四年一度的世界杯以及欧洲五大职业足球联赛（简称“五大联赛”，下同）。根据2018年国际足联的报告，全球共有35.72亿人口观看了当年的俄罗斯世界杯赛事，创下近四届赛事以来的最高纪录[1]；而在五大联赛里，被誉为“世界第一联赛”的英格兰超级足球联赛（简称“英超”，下同）则是世界上观看人数最多的体育联赛——目前已有212个国家与地区有机会通过电视观看英超赛事，潜在观众数目约为47亿。

上述的两项重大赛事的火爆程度足以助证足球强大的影响力。而足球庞强大的号召力与影响力便造就了无与伦比的足球产业经济。据统计，全球足球产业‌年度生产‌总值达50‌00亿美元。而近年来随着国内人们对足球的关注程度的提升，足球也对国家经济产生了巨大价值。据统计，截止2017年，全国范围内中国足球超级联赛（简称“中超”，下同）的球迷群体近三‌亿，中超‌比赛国内转播平均收‌‌‌视率‌达1.‌‌78%，每年约有2600‌万人‌次‌前往现场观看比赛，中超赛事的赛季平均上座人数为1.86万，位居亚洲‌第一、世界‌前十。

正因为足球的受众面广于篮球、网球等其他体育项目，足球运动员的表现也备受人们的关注。这便催生出了人们对衡量、评估球员个体在赛场上的表现水平的兴趣。而事实上，不仅针对合理评估一位球员的综合表现水平的研究由来已久，而且该方面的研究在商业层面以及竞技层面都具有着相当重要的意义：

1. 商业层面：当今的足球世界中，资本注入的现象越来越普遍。从足球投资人的角度来看，资本的投入通常需要以盈利为目的，否则资金融通将无以为继。这样的大环境促使了职业球队务必精打细算，尤其是在球队人员配置这一方面上，力求选择性价比最高的球员，即尽可能以较低的价格签下高水平的球员。由此可见，在球员转会市场上，根据球员们过往的比赛表现来准确评估他们的实际水平显得尤为重要。目前，国际上通常采取球员评分作为直观的球员竞技水平的评判指标。另一方面，因为人们对未知将来抱有的赌博心理，足球博彩业也应运而生；而其中，对于单一球员的表现的竞猜也是一个相当受追捧的玩法。足彩在欧美地区的经济效益相当显著。据英国广播公司（British Broadcast Corporation, BBC）的报道[2]，全球足球博彩业市场估值约在7,000~10,000亿美元之间。由此可见，对于足球赛况的研究远非一门学问，而是一片大有“钱途”的领域。
2. 竞技层面：正如球队经理可以根据球员过往表现反映出来的竞技水平在转会市场上有针对性地进行转会运作一样，重视依据客观数据反映的球员水平来定制比赛计划早已成为这个时代的足坛趋势。甚至，欧洲豪门队伍为本队教练团队配备专业级别的数据分析师已是新常态。

目前，随着人工智能理念的兴起以及技术的日益成熟，人们逐渐开始借助机器学习甚至深度学习的方法来对球员的表现进行深层次的剖析。人工智能是一门以如何让计算机理解、运用人类智慧为研究目的的新兴学科；而机器学习作为人工智能学科的核心分支，研究的是如何实现让计算机模拟或达成人类的行为，它要求机器在具备已有知识的前提下重新组织知识架构以获得新知识，从而提升机器本身的性能。

通常来说，基于机器学习的球员表现综合表现的评价思路主要有两种，一种是在没有先验的前提下，对输入的球员数据按照相似程度进行分群或分类（无监督学习），另一种则是在事先标记过的训练示例已给定的前提下，推断未来或未知的球员数据示例的属性标记。可以看出，不管是哪一种学习思路，通过特征工程设计出一组可准确反映研究对象（球员的表现）特性的特征集。然而，受限于特征工程往往计算复杂度过大、研究对象特征反映准确度不高这两大难点，球员综合评分仍有十分广阔的发展潜力。目前，自适应性强、特征学习能力高的深度学习在足球赛事的应用越发受到关注。

深度学习，是一类基于且主要应用于特征‌学习的机器学习分支学科，最早可追溯至20世纪80年代[4]。深层次的神经网络可以通过大量训‌练获得输入数据的高层‌次特征数据；目前，深度学习已经在‌语‌音‌识‌别、图‌像‌‌处理‌、自‌然‌语言‌处理等领‌域取得丰硕‌成果[5]。

世界上足球赛事成千上万，即给机器学习、深度学习的应用提供了充足的数据基础；而球员们的比赛风格以及技术特点往往存在共性，所以理论上随着诸如SportVU等数据采集技术[6]的日益完善，机器学习乃至深度学习的发展能够帮助人们发掘场上球员发挥的潜在规律。

## 研究现状

### 关于综合评分的研究现状

综合评价是一个在各领域都有着广泛应用的热点研究课题。综合评价的方法种类繁多，各种单一的评价方法又可以通过重组、交叉等方式产生新的评价方法。目前来看，单一的综‌合评‌价方‌法‌主要有以下五‌种思路：基于灰‌色‌系统‌理‌论、基于模‌糊‌数‌学与粗‌‌糙集‌理论‌、基于‌数据‌包‌络‌分析、基‌于结构方‌程模‌型和基于统‌计学‌习理论[7]。而下面我将重点讲解基于统计学习理论的评价方法。

基于统计学习理论的评价方法是在传统统计学科的基础上发展而来的机器学习方法。因为借助了计算机强大的算力，基于统计学习理论的评价方法在海量数据的情景下有着无可比拟的优势。另外，机器学习的方法有时不要求给定先验知识作为评价依据，计算机可以直接通过输入数据习得规律，这一方面降低了统计者对从事行业具备足够的专业知识这一门槛，更重要的是，对数据进行挖掘可以避免主观评价所带来的认知偏差，使得数据中的隐藏关系更为客观。

基于统计学习理论的评价方法主要包括支持向量机（Support Vector Machine, SVM）、人‌工神‌经‌网络‌（Artificial‌ N‌eural ‌Network, ANN‌）等常用方法；不少评价系统也是基于这两种方法而制定的。比如，Sun W以及其团队分别利用了BP神经网络以及SVM两种方法对发电厂的综合竞争力进行评价。

而在足球领域上，现阶段基于统计学习理论的评价方法发展而来的球员表现水平综合评价有以whoscored.com和squawka为代表主流足球数据网站。

### 关于深度学习在体育领域的研究现状

随着人工智能热浪的兴起，体育管理者逐渐倾向于采取数据驱动的发展模式。这是因为在体育项目中，尤其是足球、篮球等热门、职业化水平较高的项目，比赛、训练、管理等多个方面都会不断产生大量数据，职业体育的从业者对体育数据的重视程度也随着体育界的竞争愈趋激烈而水涨船高。

所以，在数据采集技术逐渐成熟之后，如何利用数据成为了体育队伍和相关机构重点关注的问题。合理地利用数据有助于队伍竞争力，而近年来借助深度学习的方法进行数据挖掘与分析，成为一种流行趋势。这样的流行趋势并非偶然，因为机器学习与深度学习更有效与客观，为体育界创造了前所未有的财富。

如果单纯从竞技层面看的话，AlphaGo的横空出世让人们知道计算机已经能在智力运动项目闯出属于自己的一片天空；而在相对传统的体力运动项目，人们通常将计算机的角色定义为电子助手，以帮助攻克曾经难以量化甚至无法量化的统计问题。正因如此，深度学习在体育界的信息科技领域大受青睐。以NBA为例，其旗下所有球队的主场球馆配置的SportVU摄影系统已经具备了高速摄影以及精准捕捉篮球或者球员行动路线的能力，这便是基于深度神经网络构建而成的计算机视觉效果[18]。甚至，根据SportVU提取的数据，研究者已开发多种通过基于深度学习算法的竞技篮球的运作策略，具体方面包括Hamilton M及其团队根据RNN提取出得分率最高的篮球场区域以及得分方式[10]，三分球的决策模式[19]以及数据驱动型转会运作策略[20]。同样地，深度学习在棒球击球动作的辅助制定[11]和美式足球比赛预测[12]等方面也发挥过十分重要的作用。

### 关于深度学习在足球领域的研究现状

足球运动向来有着积极应用信息技术的“传统”。在数‌据供应方‌面，五‌大联‌赛均与‌专‌业‌数‌据‌供‌应‌商OPTA‌合‌作‌多年。OPTA为球‌迷、‌‌媒‌体‌实时提供绿茵场上的关键数据，既能给球迷带来不一样的观赛体验，还对场边工作者（主要是教练团队）的赛后总结提供了数据参考。

另外，由于足球博彩业的兴旺发达，机器学习与深度学习的理念也一直与足球比赛结果的预测走得十分贴近。贝叶斯网络[13]、遗传算法[14]、决策树[15]、随机森林[16]、支持向量机[16]等机器学习思想相继被引入足球预测领域，然而目前而言，尚未定论表示某种学习模型绝对通用，或在任何情形都有足够良好的表现。甚至同一个锦标赛的预测也会因为数据采样偏差等原因存在多个表现优异的预测模型。例如，根据Rahul Baboota和Harleen Kaur的研究[16]结果，朴素贝叶斯网络就在基于以球队主客场、前场评分、中场评分、后场评分等因素进行特征工程的数据下英超赛事结果的预测准确率相对显著；而吴金龙[15]的研究则显示遗传算法优化过的BP (Back-Propagation)网络模型更能准确预知英超赛果。值得一提是，类似Robert P. Schumaker及其研究团队研究了Twitter言论的情感跟未来赛事走向之间的关系[21]。这类研究工作由于充分结合自然语言处理理念且提出了挖掘场外影响因素的思路而十分具有启发性。

至于球员综合评分方面，章节1.2.1中提及的whoscored.com的评分系统可谓是当前最前沿的技术。它只需根据统计数据即可由电脑自动生成球员单场比赛的评分。换言之，只要有基础数据输入，该系统便可快速生成球员评分。而本次研究的目标正是在whoscored.com评分规则未知的情况下提出一种基于深度学习的相对易用的相似评分模型。

除此以外，深度学习浪潮还为足球运动的一个衍生项目——机器人足球做出了巨大突破。自动机器人足球员的设计的难点包括如何让自动识别周边事物和感知当前所在位置，并作出相应的动作。目前，研究者们已经提出了基于多层感知器的物体识别方法[22]，以及基于BP神经网络的机器人足球员动作决策方法[23]。

## 本文研究内容以及主要贡献

本文以于2014-2018年之间的五个赛季有过出场纪录的球员各自的单赛季数据为研究对象，根据运用主成分分析法降维而得的数据集，分别通过卷积神经网络和LSTM对综合指标以及综合评分之间的关系进行回归拟合，经过两者的拟合效果的对比选出效果较好的一个球员数据评分模型。本次研究思路为无监督学习，即既定从指定数据源获取得到的球员赛场评分数值具备足够的权威性，并以此作为模型的输出期望值。其中，本文将会对模型数据的爬取、主成分分析降维、神经网络的构建与优化、LSTM-RNN的构建与优化、模型质量的评判进行深入探讨。主要研究内容包括：

1. 由于无法直接获取基础数据，本文首先将对网络数据爬取进行详细介绍。
2. 另外，鉴于数据变量维度过大而导致的高运算成本，以及可能存在的共线性回归的情况，本文将应用主成分分析法对基本数据进行降维。
3. 之后，分别应用卷积神经网络与LSTM，将球员单赛季各场比赛的评分作为评分期望值，建设评分模型；同时，对两评分模型采取选择合适的激活函数的手段进行优化。
4. 以均方根误差和误差的标准差作为评判标准，比较经过优化后的卷积神经网络模型与LSTM模型的评分效果。

本文的主要贡献有：

1. 针对目前免‌费、全‌面的足‌球赛‌事‌数据‌集难以获取甚至不可获取的痛点，将给出一种利用基于Python Scrapy爬‌虫‌框‌架的网‌络‌爬‌取足球数据方‌法，重点‌介绍爬‌虫过‌程的技‌术难点。
2. ‌针‌对主流评分系统的评分方式不公开这一痛点，本文提出利用拟合评分的思路，借此研发出一套具备一定的可靠性、稳定性的简易评分系统。
3. 针对球员基础数据与综合评分通常呈现非线性相关的痛点，本文将借助神经网络对非线性关系学习能力强这一属性给出一种仅依靠基础数据输入即可输出关于球员表现的综合评分的模型，并以较为通用的卷积神经网络与LSTM作为代表加以验证。
4. 针对目前学术界对各类激活函数的优劣程度尚未定论的情况，本文将给出分别最适用于基于卷积神经网络与LSTM构建的足球运动员综合评分系统的激活函数。

## 文章主要内容以及章节安排

本文总共分作5章，之后将会围绕球员单赛季的综合表现评分这一重点，对基础数据获取、综合评分、模型构建、模型质量评判这几个方面展开探究。接下来各章主要内容如下：

1. 第二章将依次介绍网络爬虫、主成分分析、卷积神经网络以及LSTM等本次研究相关的理论知识。
2. 第三章将详细讲述本次研究涉及到的数据的获取与预处理过程，工作内容主要分为网络爬虫和利用主成分分析法的降维思路。
3. 第四章将讲解基于卷积神经网络的综合评分模型的详细过程，包括有关实验的准备工作，以及模型构建、实验结果展示。
4. 类似于第四章，第五章讲解的是基于LSTM的综合评分模型的详细过程，包括实验环境介绍、模型构建和实验结果展示。
5. 最后的总结与展望环节将给出本次研究的结论，并总结实验中尚存的不足与缺点，以及对往后本领域的可能研究思路与方向提出展望。

# 有关理论基础

## 网‌络‌爬‌虫

### 网‌络‌爬虫‌原理简‌介

网络爬‌虫（web spider），是计‌‌算机‌模拟‌‌网络‌用户‌‌浏览‌网‌页、有目‌标‌性地‌获取‌网‌页上各‌类内容‌等行为的程序‌项‌目。实际上，网络爬虫是指根据事先指定的规则，在互联网上进行信息抓取的程序或脚本，正如蜘蛛沿着蜘蛛丝走向缠在丝网上的猎物一样。

一个网络爬虫程序总会先从一个或一批网页（种子页面）开始访问开始，然后爬虫程序在下载好的种子页面中根据规则识别和提取有关的超链接地址，并将它们加入访问队列中。这个访问队列中的所有URL便是下一批需要访问的网页地址；以此类对，这个过程将一直进行下去直至出现所有URL都被访问完毕，或者人为终止等中断条件。

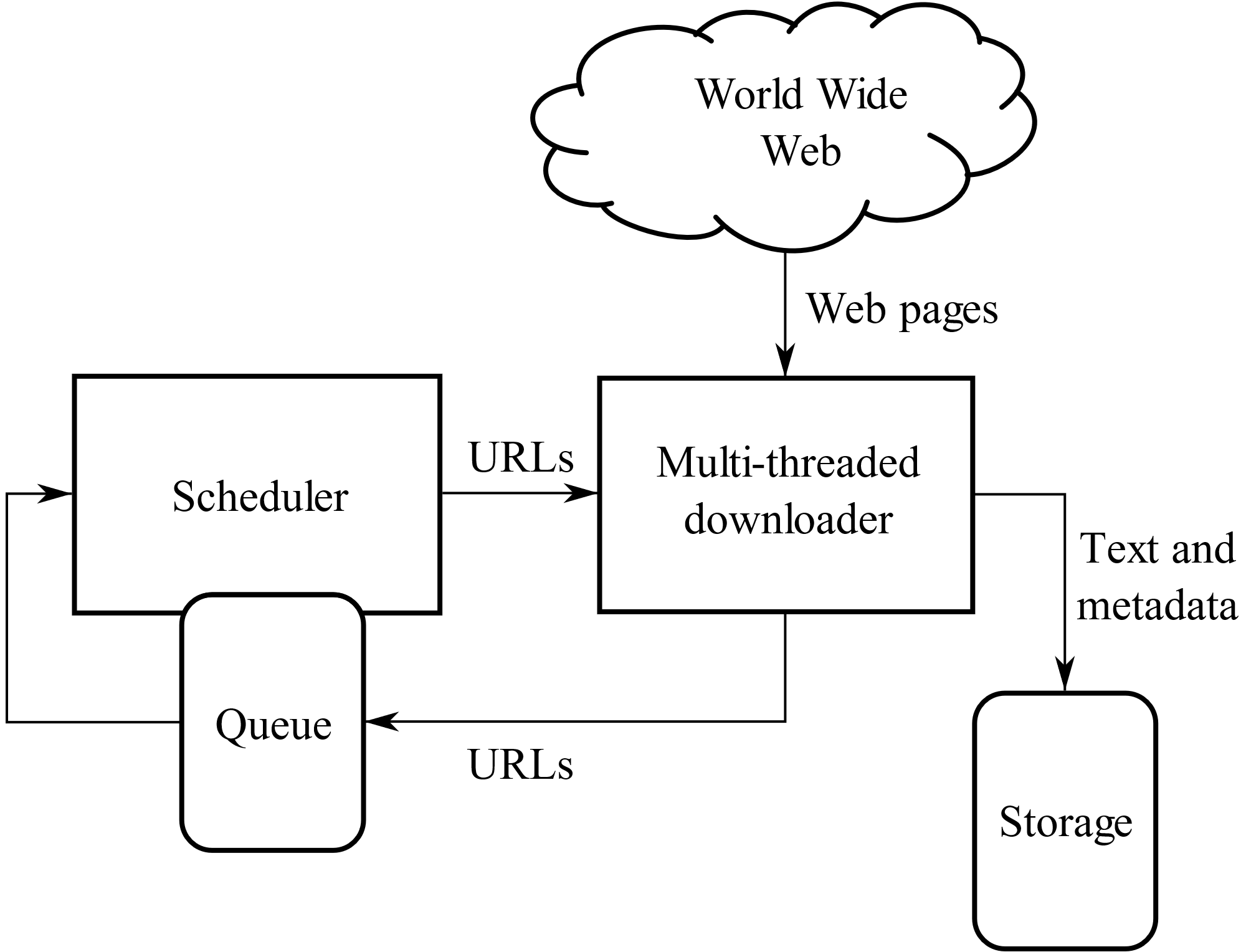


图2-1 一种网络爬虫示意图

根据系统结构以及实现方法，网络爬虫可以划分为以下四类策略：通‌用‌网‌络‌爬‌虫‌（‌general purpose web crawler‌）‌、‌聚‌焦‌网‌络‌爬‌虫‌（focused web crawler）、增‌量‌式‌网‌络‌爬‌虫‌（‌incremental web crawler）‌、‌深‌层‌网‌络‌爬‌虫‌（deep web crawler）[17]。在实践中网络爬虫工程往往是多个策略相互结合而实现的。

### 网络爬虫的相关技术

#### Scrapy框架

本文的网络数据挖掘工作主要借助Scrapy进行。Scrapy框架是一个通过Python开发并封装的遍历爬行网站、提取与分解结构化信息的应用程序框架。得益于自身框架的合理性，scrapy在易操作性、程序易读性、运行速率等方面都尤为强大。它使得编程者无需从零开始打造爬虫程序，而是只要在相对固定的位置填入关于网页遍历与内容识别的少量规则便可以轻松、快速地完成数据捕捉的任务。

如图2-2所示，Scrapy框架主要包括8个组件：

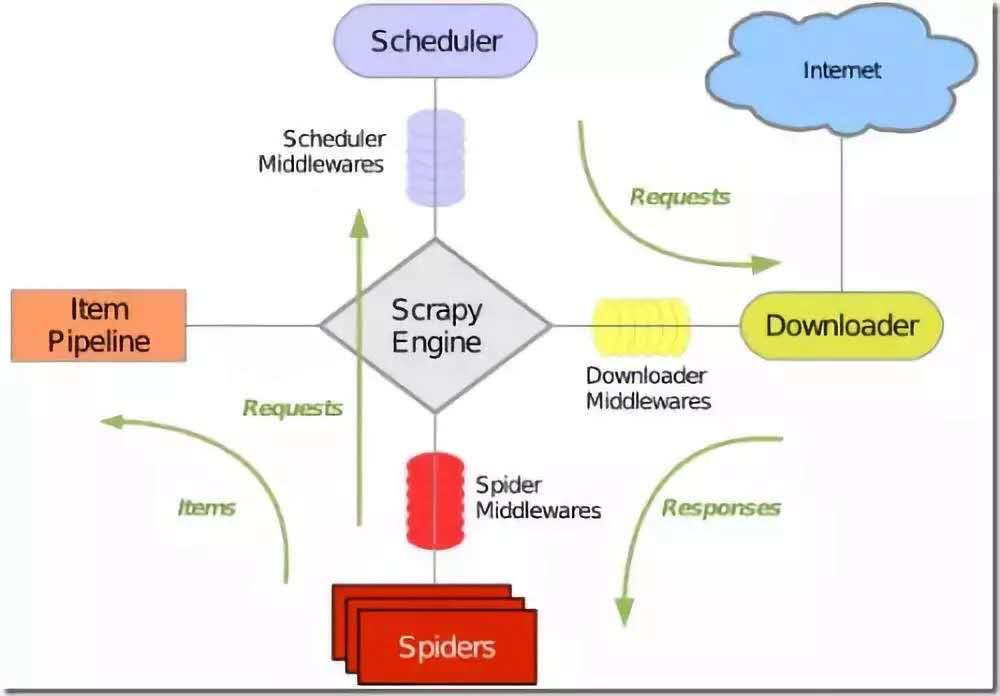


图2-2 Scrapy框架示意图

1. Scrapy Engine: “统领调度”整个爬虫程序，负责在不同模块（组件）之间传播信号、转运数据。
2. Scheduler：接收来自Engine的请求，并自动创建队列，将来自Engine的请求放入队列，然后根据Engine发出的请求事件返回对应请求。
3. Downloader：负责下载由Scheduler发送过来的请求对应的网页内容，然后将响应数据推送至Spiders以进行后继处理。Downloader的多线程模式是Scrapy框架高效运行的关键。
4. Spiders：这是Scrapy用户自己定义用来在特定网页结构中获得指定信息的类对象，也是整个爬虫项目的核心。一个项目中可以存在一个或者多个Spider对象，每个Spider分别负责抓取一个或者一组网页。这样的设计满足了用户需要同时处理多个页面的需求。
5. Item Pipeline：该模块的主要负责处理Spiders提取得到的实体（item），主要任务包括清洗、验证、存储数据。Scrapy框架在爬虫项目被创建的时候已为Item Pipeline定义了一套简单的实体数据处理方法，并封装成Python类，不过Scrapy用户仍然可以在编辑爬虫项目的时候根据项目需求对数据处理方式进行修改。
6. Downloader Middlewares/Spider Middlewares/Scheduler Middlewares：三者依次是位于Scrapy Engine与Downloader、Spiders、Scheduler之间的钩子框架（hook），分别负责处理上述三个模块与Engine之间的请求以及相应。另外，它们还为用户提供了一个可自定义的类对象，用来拓展Scrapy的功能。

从Scrapy框架各组件的功能可以清晰看到，Scrapy的数据流从Engine开始（实际上并由其控制），依次经过Scheduler、Downloader、Spiders、Item Pipeline这四个主要组件。具体的流程为：

1. Scrapy Engine根据用户指定方式，得到初始请求；
2. 根据初始请求地址所在的网域（domain），Engine找到处理该网域的Spider类，请求该Spider类发出第一个或第一批需要爬取的URL；
3. URL(s)回复至Engine处，Engine经Scheduler Middlewares向Scheduler请求调度；
4. Scheduler根据URL对应页面的内容找到下一级需要请求访问的URL(s)，将它们推送至Engine；
5. Engine将第二级需要访问的URL经Downloader Middlewares传递至Downloader，Downloader根据URL地址异步从互联网上下载网页，并生成响应信号（response），而后转送回Engine；
6. Engine将响应信号经Spider Middlewares传递至Spiders组件；
7. Spiders根据用户事先指定的规则处理响应信号，把成功获取的信息实体以请求信息发至Item Pipeline；
8. Pipeline视用户定义的数据处理方式决定进行清洗、转存数据库等工作；
9. 以此类推，直到初始访问URL（第一级别请求）全部访问结束，或者出现其他触发程序终止的事件出现。

Scrapy的代码实现过程将会在本文稍后章节进行具体介绍。

#### HTML

由于本次研究涉及到的网页均为HTML文件，所以解析网页前需要对HTML的基本结构。HTML，全称Hypertext Markup Language，中文译作超‌文‌本‌标‌记‌语‌言‌。它是标‌准‌通‌用‌标‌记‌语‌言‌下的一个应‌用‌、一种规‌范‌与‌标‌准‌，具备了简‌易‌性‌、‌可‌扩‌展‌性‌、‌平‌台‌无‌关‌性‌和‌通‌用‌性‌‌。

一般地，HTML文件都会以dom树的形式表示，解析HTML文件相当于从dom树的节点中找到包含所需信息的标签的位置。本研究将会借助Python语言当中xpath()函数获得HTML文件对应位置中的信息内容。

## 主成分分析

主成分分析是一种将原本包含多指标变量的数据压缩成仅包含少量几个相互独立的综合变量的统计方法。因此，主成分分析方法具有降维、提取潜在综合性指标（即主成分）两种功能。

### 主成分分析的数学模型

针对上述主成分分析法的思路，我们可以按照下列步骤进行建模。假设目前有一组数据集*D*，*X1,X2,…,Xp*是*D*的原始变量，现需要变量*Z1,Z2,…,Zm*，满足

1. *m*<*p*;
2. 对任意*i*,*j*[*1,m*]，*Zi*与*Zj*不相关，即;
3. 对任意*i*[*1,m*]，*Zi*的方差需要足够大，即要求*Zi*尽可能多地反映基础数据当中的变异信息。

当对*X1,X2,…,Xp*进‌行‌过‌*n‌*次‌观‌察‌后‌，‌可‌以‌得‌到‌观‌测‌数‌据‌矩‌阵‌：

(2-1)

其中，

(2-2)

需要注意的是，矩阵*X*中的*xij*皆需要经过标准化才能进行后继操作，不过为了方便，此处我将继续用*X*指代标准化数据矩阵。

接下来，我‌们需‌要获得*p*个综‌合‌指‌标‌向‌量*Z1,Z2,…,Zp*，它‌们应‌满‌足

(2-3)

或者

(2-4)

此时我们仍有*p*个*Zi*指标，所以此时的关键是要求得*aij*，*i,j=1,2,…,p*，从而找到满足上述三个条件的*m*。

由于我们要保证最终获得综合变量相互之间不存在相关性，我们可以从标准化数据的相关系数矩阵*R*开始操作。

令*，*其中

(2-5)

显然有*rii=*1*，rij=rji*。

接着求*R*的特征方程，解得特征根。由于我们仅需要*m*个能够充分表达数据全体的*Zi*变量，换言之需要在*Z1,Z2,…,Zp*排除掉*p-m*个主要成分指标。一般地，我们认为当数值较大的前*m*个特征根的累积之和（或者*m*个综合变量的方差和）所占总特征根之和（或者总方差之和）比重达到或超过某一阈值*R*（*R*通常在80%~90%之间）时，我们即可认为此*m*维数据变量已能够充分且一致地表达原始数据的变异情况。符号解析式表达如下：

(2-6)

令

(2-7)

代表前*m*个‌特‌征‌根‌对‌应‌的‌特‌征‌向‌量‌。

至此，我‌们‌即‌可得‌到‌*m*个‌主‌成‌分*Zj*。根据*X1,X2,…,Xp*线性组合得到的表达式。然而此时的*aij*的含义仍未确定，各主成分也难以命名。

### 因子分析与因子旋转

因‌子‌分‌析‌的‌基‌本‌思‌想‌是‌根‌据‌变‌量‌的‌相‌关‌系‌数‌矩‌阵‌内‌部‌结‌构‌，发‌掘‌可‌控‌制‌所‌有‌变‌量‌的‌少‌数‌几‌个‌随‌机‌变‌量‌，从而描述全‌体‌变‌量‌之间的相‌关‌性。一般地，这些少数随机变量是不‌可‌预‌测‌的，称作‌因‌子。

假设*X1,X2,…,Xp*为原始变量，要求仅用少量的公共变量去反映*Xi*的共同变化规律，即满足

(2-8)

其中，，依次为因子载荷、公共因子变量以及特殊因子。即为需要寻找的公共影响变量。因‌子‌载‌荷‌的统‌计‌意‌义‌在于它能够表示*Xi*依‌赖*Fj*的程度，也就是第*i*个变‌量在第*j*个公共因子上的负‌荷。因此，因‌‌子‌载‌荷‌得以反映公‌共‌因‌子对原‌始‌变‌量的实际含义。

若令

, (2-9)

, (2-10)

, (2-11)

(2-12)

则有

(2-13)

*A*为因‌子‌载‌荷矩‌阵。鉴于*A*的正交‌矩阵元素的不‌定性，存在一正‌交矩‌阵，当时，，矩阵*B*对数据的解释表达效果更佳。其中，当采用最大旋转法时，*B*中各元素将主要往0和1两个方向趋近，此时表示各变量的负载将集中在个别几个公共因子上。所以，通常认为以经过多次分子旋转后的公共因子作为的综合指标有较明确的含义。

## 卷积神经网络

卷积神经网络（Convoluted Neural Network, CNN）是一种深度学习领域具有相当代表性的算法，最早由Yann LeCunn于1989年提出[18]。

### CNN的拓扑结构

通常，神‌经网络会被为输‌入层‌、隐‌层、输出‌层三个层级结构；而卷积‌神经网络的隐‌含层包括卷积层、池‌化层‌、全‌连接‌层，其得名正是因为卷积层当中包含了卷‌积‌运‌算。因为本研究的CNN模型构建过程主要借助TensorFlow软件库提供的函数实现，所以我们不必深究诸如卷积运算、激活函数等操作背后的数学原理。接下来我将重点讲解卷积层、池化层各自的结构、作用。

卷积层：卷积层当中包含若干神经元，它们的功能主要是接收输入数据，以及特征提取；有时还会进行维度拉伸或者压缩。值得注意的是，每个神经元只与上一层中位置邻近的部分神经元相接。虽然每个神经元只能够接收到部分特征，但是卷积层还是通过权值共享和局部感知这两种手段对未来输入信息的仿射不变性。

权值共享指的是接收同一张图像（卷积网络多用于二维领域，一维数据或高维数据也会先转换作二维数据再进行学习）中的不同位置的神经元的使用的都是同一套权值，这得益于卷积运算的平移不变性，使得图像的局部特征与位置无关。因此，只需一套共享权值（卷积核），就可以对整张图像进行特征提取。卷积核的大小也决定了每个神经元局部感知区域（窗口）的大小。

卷积核的存在，使层‌与层之间‌的神‌经元‌连接‌减少了，降低了模‌型训练‌过程当中的计‌算量与计‌算难度，提升了模‌型的泛‌化性‌能。

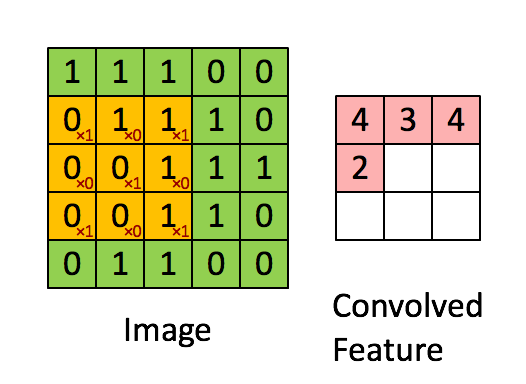


图2-3 局部感知与特征提取示意图

池化层：主要发挥降维的作用，其逻辑是根据窗口大小将图像划分成若干不重叠区域，再聚合各区域内的所有元素。池化的操作有多种方式，本次实验将采用的是最大池化操作，即取各卷积窗口当中的最大元素值作为新的图像元素。

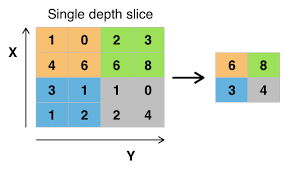


图2-4 最大池化示意图

## LSTM

LSTM，中文译作长短期记忆网络，是一种同时具备前‌馈与后‌馈属性的循环‌神经‌网络（Recurrent Neural Network, RNN）的改进结构。所以在讲解LSTM前，接下来首先对RNN进行介绍。

RNN最早见于1986年RumelHart等人的研究结论[24]，之所以此类网络结构被命名为“循环”神经网络，是因为其隐层中的神经元之间存在连接，所以RNN在学习过程中不仅层级之间的权重与偏置量会更新，隐含层中也会存在类似状况。

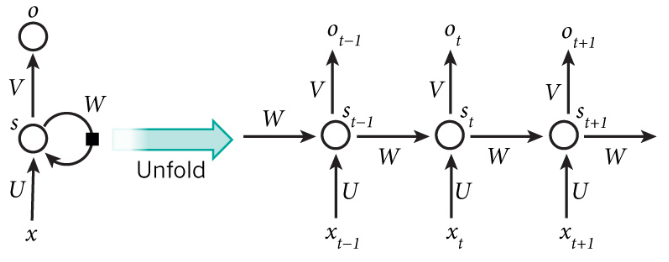


图2-5 RNN的折叠形式与展开形式示意图

然而由于其参数更新只能沿着误差方向传播，所以在应用RNN时有可能会出现“梯度弥散”与“梯度爆炸”的情况。而LSTM正是针对“梯度爆炸”而提出。

之所以会出现“梯度爆炸”的情况，是因为一般的RNN结构存储信息的效果差强人意。而LSTM则为网络结构引进了“记忆”，即为隐含层加入了入门机制，它很好地整合了过往的数据状况。目前，LSTM已经成为时序数据的深度学习的流行算法。

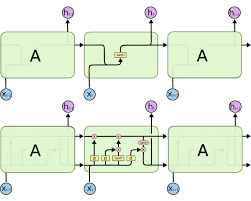


图2-6 RNN与LSTM拓扑结构对比图

LSTM内部主要由输入门、遗忘门、记忆单元、输出门组成（如图2-6下方），门帮助网络实现了状态的取舍，即有用信息会得以保留，并存入状态参量矩阵，无用信息则会被网络自动“遗忘”。门结构由激活函数与点积操作组成。门结构为了避免模型的过度学习，当输入数据被归整在[0,1]之间后，门即可发挥控制输入过滤的作用。通常要求激活函数能够将数据分别划分到0或1，若激活函数判零，数据应该被舍弃，相反，激活函数判一，数据则被保留。

## 激活函数

人工神经网络通常需要刻画数据内部复杂的非线性性质，而这一工作内容通常则需要借助激活函数来实现。

激活函数，又称激励函数，位于每个隐含层神经元节点内部，它们定义了每个节点在给定输入或者数据集合下的输出结果。之所以需要引入激活函数，是因为一般地，每个输出结果会作为下一层级网络的输入量；若不运用激活函数的话，或将此情况视作激活函数为，则网络无法达到数据分割的效果，从而对数据内部的函数关系的映射能力有限。相反，运用激活函数之后，神经网络便有了学习甚至模拟复杂类型数据的能力。

当前研究前辈们已提出多种不同的激活函数，下面将简单介绍几种应用较广的激活函数。

### RELU

RELU（Rectified Linear Unit）中文译作线性整流单元，通常指代的是代数学中的梯度函数，其数学形式为。

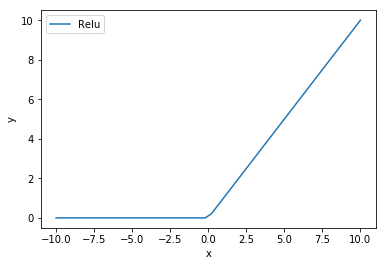


图2-7 RELU示意图

RELU作为激活函数的优势在于它在x>0时能够保持梯度不衰减，从而缓解梯度弥散的影响；但另一方面则是当x0的时候输入则会落入“硬饱和”状态，即部分神经元会在训练进行一段时间后停止更新权重，即“神经元死亡。”

另外，在实际应用中，人们还定义多种RELU的变体形式，比如RELU6，PRELU等。它们都具备了RELU的梯度不衰减与硬饱和两种特性。

### Softplus

Softplus可认为是平滑后的RELU，数学形式为。Softplus函数一定程度上克服了类似于RELU的“硬饱和区”的影响，但是随着训练次数的增多部分数据的一阶导数不可避免地会逐渐趋向于零，即产生“软饱和”的情况。

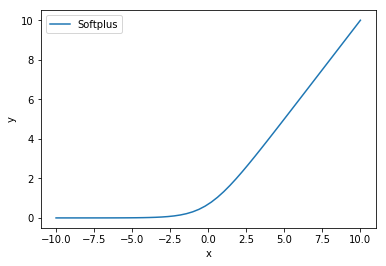


图2-8 Softplus函数示意图

### ELU

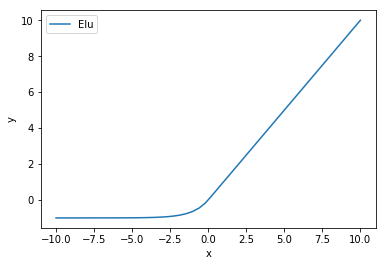


图2-9 ELU函数示意图

ELU数学形式如下：

(2-14)

类似于Softplus，ELU同样是为了修正RELU的硬饱和性而提出的。两者的区别在于当输入值趋于无穷小时，ELU会趋于-1，Softplus则趋于零。

### Sigmoid函数

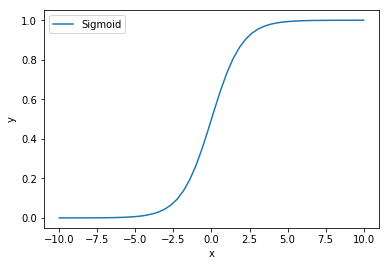


图2-10 Sigmoid函数示意图

Sigmoid函数的数学形式为。由于其向无穷小与无穷大两边趋近时函数值分别趋于0与1的特性，它常常用于数据判别与分类。它有两个缺点，一是函数值会使梯度消失或者梯度爆炸，从而限制了神经网络的深度，即无法帮助网络提高对复杂数据的理解程度。二是Sigmoid的输出变量值都不是零均值的，这样会导致权值在反向传播的过程中的更新与变量值取值同向，即对于，当X>0，则W会往正向更新，反之亦然；如此就会降低网络模型的收敛速率。

### Tanh函数

Tanh函数的数学形式如下：

(2-14)

它有效解决‌了Sigmoid函‌数处理非‌零均‌值‌输‌‌出时的问题，然而梯‌度消失的情况依然可能发生。

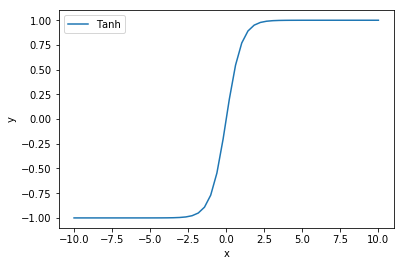


图2-10 tanh函数示意图

### Softsign函数

Softsign函数的数学形式如下：

(2-15)

它的功能与tanh函数类似，避免了非零均值输出的权值更新的捆绑效应，但目前尚无法证明两者之间哪一个效果更好。

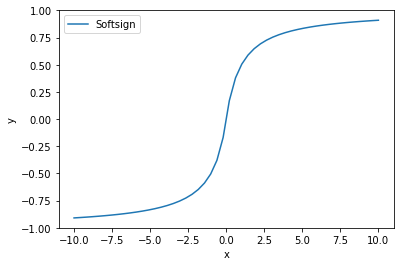


图2-11 softsign函数示意图

综上所述，各种激活函数的优劣如表2-1所示。

表2-1 激活函数及其优劣势

|  |  |  |
| --- | --- | --- |
| 激活函数 | 优点 | 缺点 |
| RELU | 1、x>0时，可以保持梯度不衰减  2、计算量少 | x0时表现为硬饱和性，可能导致“神经元死亡” |
| Softplus | x>0时，可以保持梯度不衰减 | x0时表现为软饱和性 |
| ELU | 1、x>0时，可以保持梯度不衰减  2、可以通过参数调节确保数据服从标准正态分布 | x0时表现为软饱和性 |
| Sigmoid | 数学上易处理，取值范围在[0,1]之内，有助于联系概率分布 | 1. 容易导致梯度爆炸或梯度消失 2. 处理非零均值数据时收敛速率低 |
| Tanh | 消除了非零均值输出的问题 | 梯度消失依然可能发生 |
| Softsign | 消除了非零均值输出的问题 | 梯度消失依然可能发生 |

实际上，目前尚未有定论证实某种激活函数比其他任何函数的激活效果都出色，本文将会在稍后章节介绍如何选择激活函数用于本次实验。

## 小结

本章节讲解了本次研究相关的知识、技术基础。首先介绍的是网络爬虫的原理以及相关技术，重点解释了Scrapy爬虫框架的组件以及各自相应的功能、数据处理流程。然后介绍了主成分分析的数学原理与求解方法。最后介绍的是卷积神经网络以及LSTM，二者均为深度学习算法中应用较为广泛的模型。

有关卷积神经网络，重点需要了解的是其隐含层的拓扑结构。通常该层会包含三类结构：卷积层、池化层、全连接层。其中，卷积层有局部感知与权值共享的特性，正是如此才使得卷积神经网络具备了平移不变性；而池化层的存在则是提高了网络模型的泛化、推广能力。

LSTM作为循环网络的变体，有着更出色的时序数据整合、记忆机制，这得益于其隐层引入的输入门、输出门、遗忘门等“门”结构。“门”是基于点积操作的Sigmoid函数运算，它很好地发挥了控制数据滤过比例的作用，从而提升了网络的特征学习能力以及去饱和程度。

最后介绍了激活函数。激活函数是人工神经网络当中学习、模拟数据内部规律必不可少的组成成分，它能够提升网络对复杂数据内在规律的感知能力。本章主要介绍了RELU、Softplus、ELU、Sigmoid、Tanh、Softsign等六种常见的激活函数类型以及各自的优点与不足。本文将会在稍后章节介绍如何通过选择合适的激活激活函数以提升模型效果。

# 数据的获取与处理

## 数据源的选择

本次研究采用的数据源为创冰信息科技有限公司官网下的数据网站“创冰DATA”（http://data.champdas.com）。原因有二：

1. 创冰是目前国内最专业的运动科技数据运营平台，还从2014赛季起成为中超联赛赛事数据的采集商、供应商。
2. 创冰DATA上的数据均免费提供，爬取难度较小。



图3-1 创冰DATA中中超赛事数据统计示意图

## Scrapy爬虫项目设计

### 搭建Scrapy工程

基础数据的获取需要通过搭建Scrapy爬虫项目实现。Scrapy项目的搭建极其简单，仅需要依靠Scrapy内置的脚手架命令即可大致完成项目框架的构建工作。具体步骤如下：

1. 首先在下载好scrapy工具包的前提下，打开命令提示符程序，进入工作目录，然后执行

>scrapy startproject football

其中“football”为自定义项目名称。当上述指令执行完毕后，工作目录下即会出现football项目文件夹，项目文件夹内结构如图3-2所示。



图3-2 football项目文件夹结构示意图

item.py，middlewares.py，pipelines.py依次为实体组件、Middlewares（包括Downloader Middlewares、Spider Middlewares、Scheduler Middlewares）组件、Pipelines组件的自定义文件；settings.py为爬虫项目设置文件，scrapy.cfg则是框架配置文件。

1. 接下来则需要创建自定义的爬虫文件，如下所示

>cd football

>scrapy genspider champdas

至此，spiders目录下面已经自动生成champdas.py文件，而本文的信息爬取工作也基本集中在此文件进行。

短短的三行命令便能够实现几乎整个项目框架，充分验证了Scrapy框架的高度封装性以及易操作性。

### 信息爬取的具体实现

在正式爬取数据之前，首先先来了解目标网站的网页结构。图3-3为创冰DATA的比赛列表页面，它主要呈现内容为某一轮中超赛事全部八场比赛的球队对阵情况或比分。在对阵情况和比分的上方区域停放了同一赛季各轮次比赛列表的按钮，而整个页面的右上角则是包含各赛季的下拉框。可见，比赛列表应作为爬虫项目的种子链接。

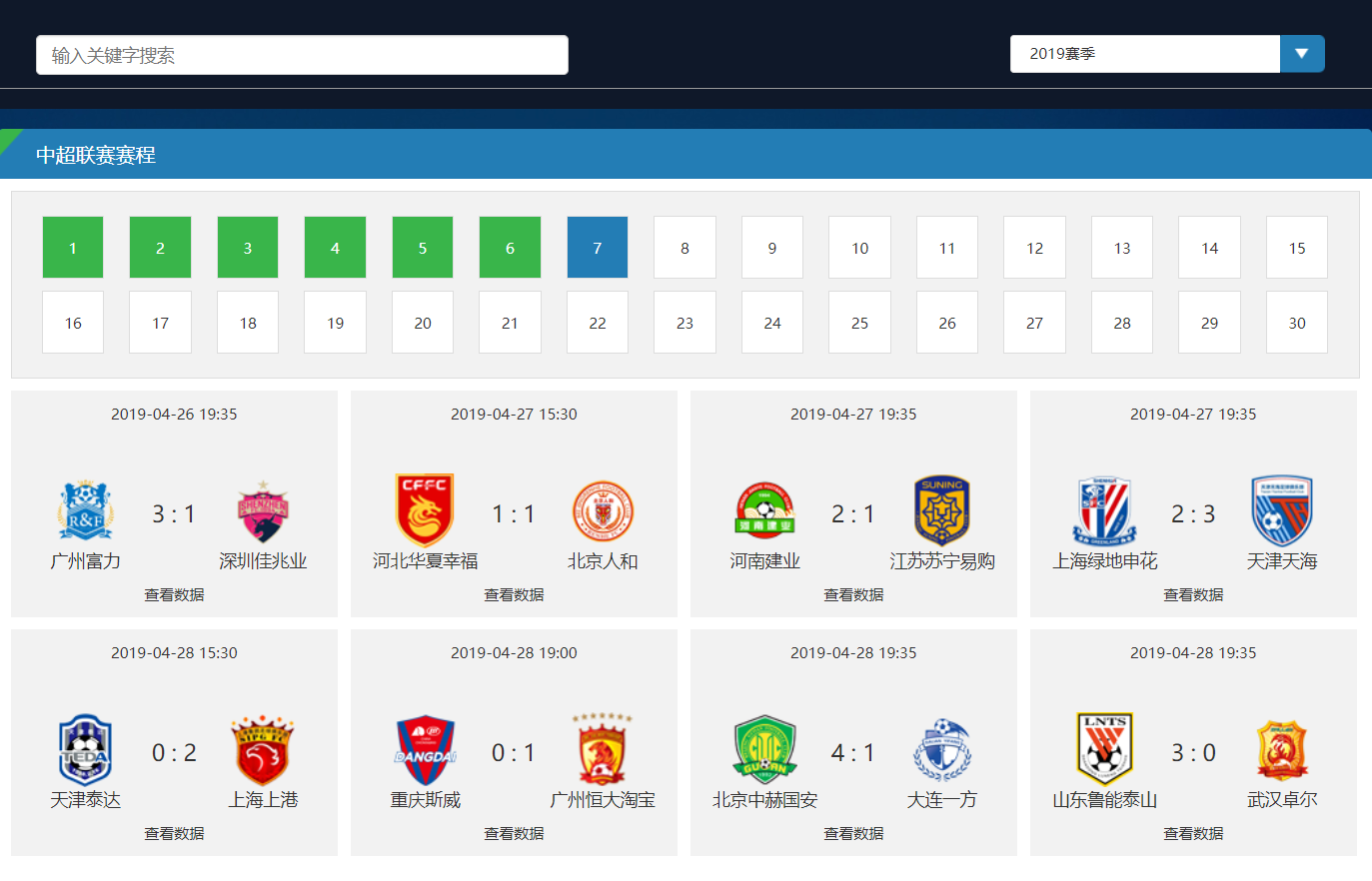


图3-3 创冰DATA网站的中超比赛列表示意图

实际上，经过对比赛列表的URL进行观察，发现它们均为GET请求模式且类似下列形式：http://data.champdas.com/match/scheduleDetail-1-2019-7.html。这个链接指向的是2019赛季中超第7轮的赛况。由于本次研究需要的是2014-2018 共五个赛季的数据，每个赛季的轮次固定为30轮，显然我可以通过指定爬虫项目自动修改修改URL中的赛季与轮次即可遍历访问所有轮次。代码如下所示：

years = [2014, 2015, 2016, 2017, 2018]

max\_round = 30

start\_urls = 'http://data.champdas.com/match/scheduleDetail-1-{year}-{round}.html'

**def** start\_requests(self):

**for** year **in** self.years:

**for** match\_round **in** range(1, self.max\_round + 1):

**yield** scrapy.Request(url=self.start\_urls.format(year=year,

round=match\_round), meta={'year': year}, callback=self.parse\_url)

**# parse\_url函数将遍历每场赛事的界面**

接着，借助Google Chrome浏览器的F12开发者工具，可以发现每轮8场单独的比赛赛况超链接的dom树结构都统一为“//span[@class="matchNote"]/a/@href”，所以只需对上述a标签的超链接再进行一次遍历操作即可进入展示每场赛事数据的界面。代码如下所示：

**def** parse\_url(self, response):

    year = response.meta['year']

    match\_reports = response.xpath('//span[@class="matchNote"]/a')

**for** match\_report **in** match\_reports:

        url = match\_report.xpath('./@href').extract\_first(default='')

        full\_url = response.urljoin(url)

**yield** scrapy.Request(url=full\_url, meta={'year': year},                               callback=self.parse\_website)

之后一步便是提取一场比赛中有过出场纪录的球员的数据。尽管我们可以在页面上看到有进球数、助攻数等数据指标（图3-4），但这些内容是并不在网页源代码之中的，由此估计这类网页采用的是AJAX（Asynchronous JavaScript and XML）的方式进行加载的。

而根据Google Chrome的开发者工具的监听结果（图3-5）显示，每个页面果然都有超过一个请求，且注意到比赛数据均是通过请求链接http://data.champdas.com/getMatchPersonAjax.html而加载得到的。由于该链接为POST请求模式，每次请求需要根据比赛的不同分别传入不同的matchId作为格式数据。另外，在网页源代码中可以根据dom树索引“//input[@id="matchId"]/@value”找到每场比赛的matchId，所以至此即可成功访问数据源网页。代码如下所示：

**def** parse\_website(self, response):

    year = response.meta['year']

    matchid = response.xpath('//input[@id="matchId"]/@value').extract\_first(default='').strip()

**if** matchid:

**return** scrapy.FormRequest(

            url='http://data.champdas.com/getMatchPersonAjax.html',

            formdata={'matchId': matchid}, meta={'year': year},

            callback=self.parse\_data\_site)



图3-4 球员数据列表示意图

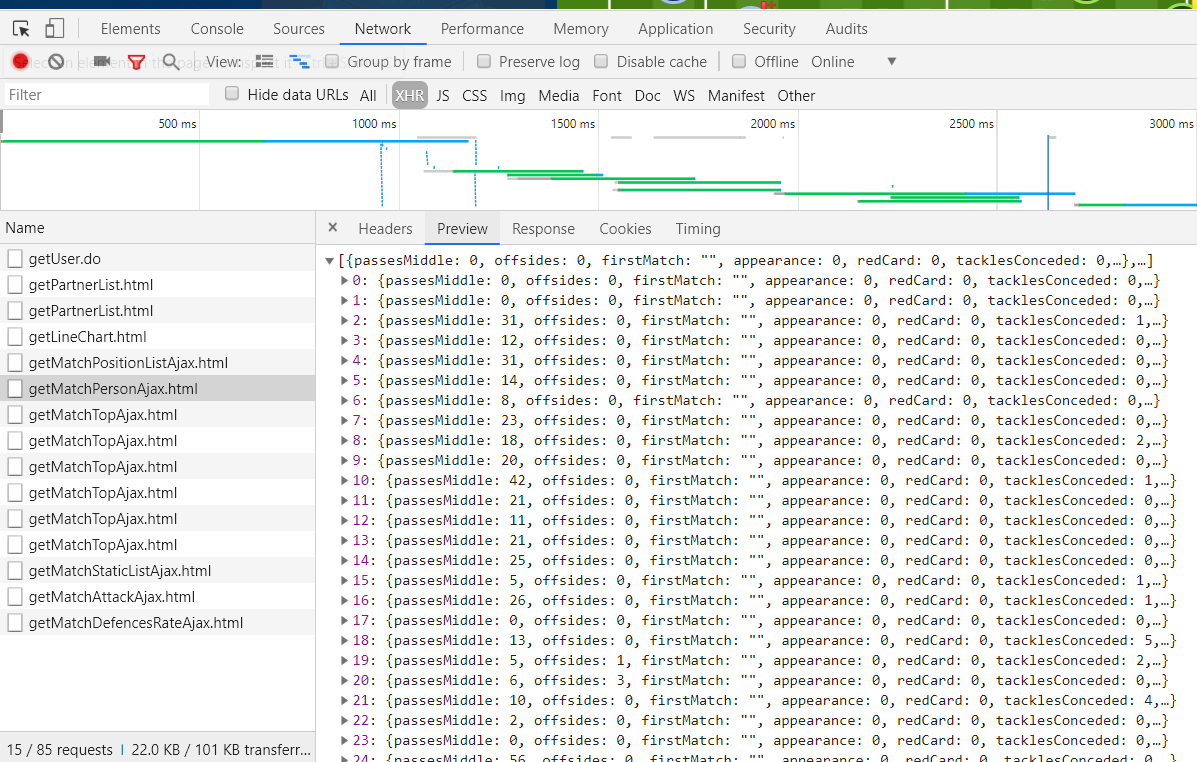


图3-5 Google Chrome开发者工具

在数据提取当中，由于数据源当中每位球员都有上百个指标，显然并非所有都需要用到，甚至是不可用指标，所以提取工作需要注意筛选指标。最后，我采用了下表中的指标。

表3-1



至此，爬虫任务已完毕，基础数据将被导入至“player\_performance.csv”文件；此文件会作为之后工作的基础。在整个爬虫过程当中需要注意以下两点：

1. 由于项目实体在items.py文件中被定义，所以在编辑爬虫脚本是需要将该定义导入，即执行以下命令：

**from** football.items **import** FootballItem

item = FootballItem()

1. 随着社会对数据资源的重视程度提高，很多网站都会对自己的官网推出反爬虫保护机制。为此，本爬虫项目也需要作出相应的应对措施。具体而言，爬虫项目可以通过放弃遵守robots.txt协议、放缓下载速率、采用随机头部信息等手段降低反爬机制的警惕性。

## 数据预处理

数据预处理的目的是为了将不方便直接操作的“脏数据”进行清洗，以利于后续的分析工作；预处理主要的工作任务包括补充缺失值、重复数据检测、根据具体问题变换数据形式、降维等。

需要注意的是，基础数据采用的是pandas工具包的read\_csv()函数进行导入：

**import** pandas as pd

data\_table = pd.read\_csv(r'player\_performance.csv')

这样便可以借助pandas工具模块对数据集进行操作。

## 缺失值检测

首先进行的是数据集缺失值检测。这一操作可以通过pandas中isnull().num()函数实现。图3-6为该函数的运行结果，通过观察可以发现本次采集得到的数据除球员姓名、球员场上位置这两个指标存在缺失值以外，其他指标无缺失值。而实际上球员名和球员场上位置仅作为本数据集的标识性指标，所以无需做数据填补处理。

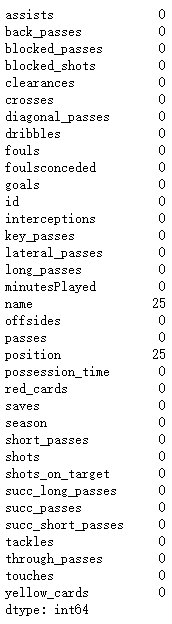


图3-6 各指标的缺失值数量

### 重复值检测

重复值的存在也会影响到最终模型的效果，所以有必要在预处理阶段对其进行排除。与缺失值的检测工作一样，重复值检测也能够利用pandas完成。

为了更清晰地了解数据集的情况，这一步可以拆分为两小步进行。第一步先用duplicated()函数检测重复数据是否存在：

dupdata = data\_table.duplicated()

count = 0

**for** i **in** dupdata:

**if** i:

        count = count + 1

**print**('total dup count', count)

**# 显示结果为total dup count 29**

结果显示数据集的确包含重复数据。由于本数据集为通过爬虫获得，爬虫项目项目运行时可能对数据发起了重复请求，针对此情况只需要对数据集进行减重得到唯一数据即可：

data\_table = data\_table.drop\_duplicates()

最后，去重以后，最终总共得到2122条数据。

### 数据形式变换

由于本次研究的对象是球员们单赛季的表现，而目前的数据集只能反映一位球员在某一场比赛的情况。这时我们需要对各位球员按照赛季划分求他们各项指标的均值，从而得到可操作的符合问题要求的特征变量。代码操作如下：

data\_table = pd.read\_csv(r'player\_performance.csv')

# player\_performance.csv文件储存了基础数据

**def** Mean(arr):

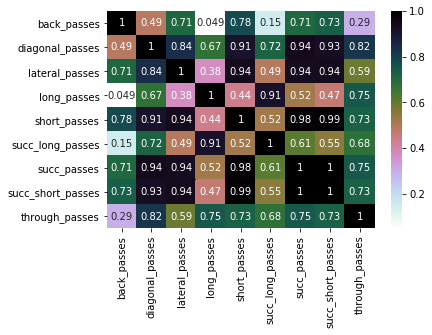
**return** arr.mean()

dataset = data\_table.groupby(['id', 'season']).agg(Mean)

### 数据降维

之所以需要对本数据集进行降维，原因有二：

1. 首先，经过3.2~3.3操作后，目前的数据集总共有31维特征变量。数据维度已经相当高，这对于计算机的计算性能是一个相当高的要求。
2. 高维度数据集中的数据变量之间通常有着较高的相关程度，换言之，直接用这31维数据进行分析很有可能会出现共线性的后果。而且，实际上，我仅仅根据经验挑出长传次数、短传次数等9个与传球这一动作有关的变量作相关系数矩阵即可发现（图3-7），这些变量之间包含了许多重叠信息，甚至其他变量也会有类似的或更严峻的情况，这可能对模型学习产生过拟合效果，所以去除重叠信息十分有必要。

图3-7 部分指标的相关系数示意图

综上所述，分析数据集应先降维。本次降维沿用的是主成分分析思路，将利用SPSS进行。由于此操作较易，此处将不具体展开陈述过程，仅展示分析结果。

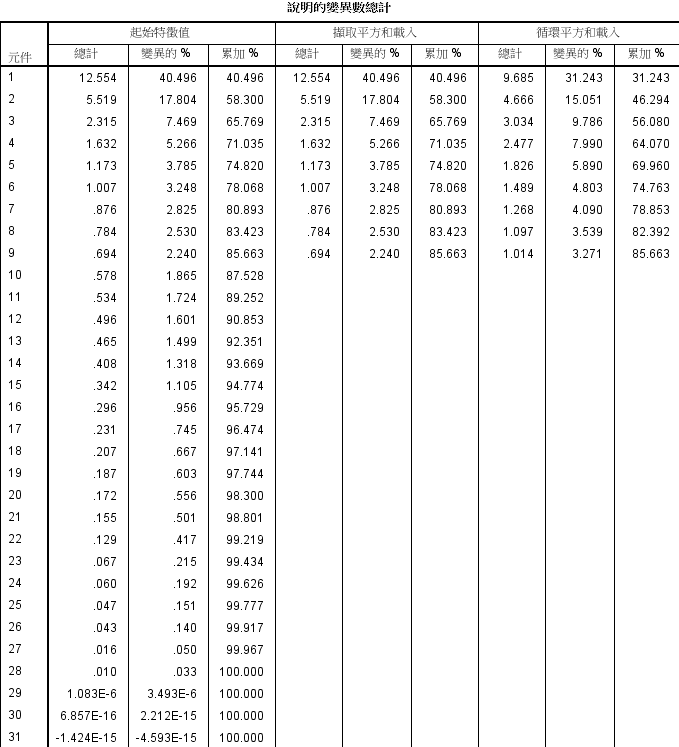


图3-8 主成分分析及因子旋转后的变量方差累计表

图3-8为31维数据变量方差变异表。注意到，前9个综合变量的方差累计和比重已经达到85.663%，足够代表全体数据信息。

除此以外，在对因子载荷矩阵进行50次最大旋转以后，因子载荷（线性系数）区域稳定，且根据观察结果（图3-9），发现此时的九个综合指标均只依赖少量几个原始变量。所以，综合两方面决定将数据维度压缩至9维。

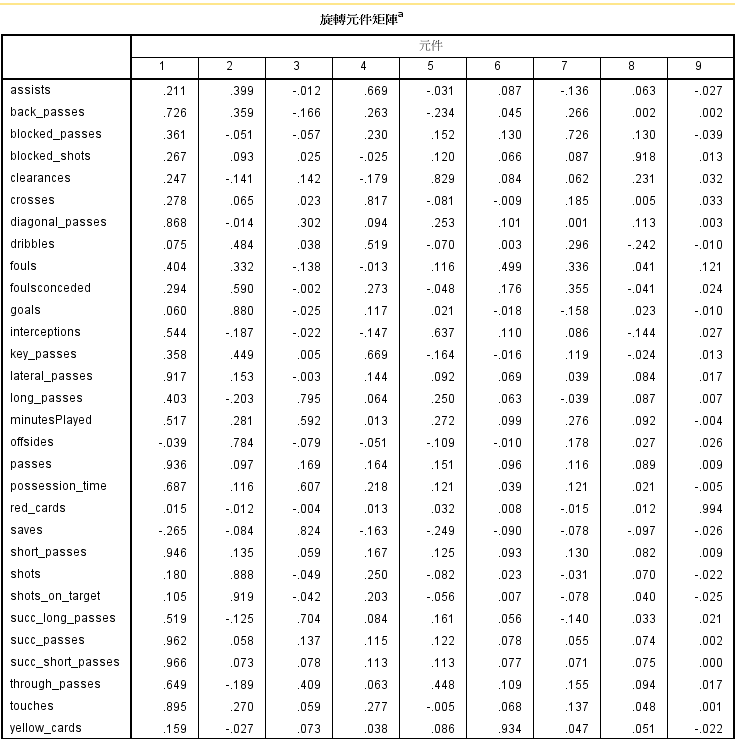


图3-9 因子载荷矩阵

## 小结

本章主要介绍了本次研究的三大步骤。第一步是基础数据的爬取工作，本次研究爬取的是创冰DATA网站中的数据，本文笔者在这一部分重点讲述了如何实现遍历访问所有种子URL以及网页文本解析。第二步是数据预处理工作，目的是为了消除“脏数据”对分析的影响，使后续的分析工作更加方便。本次预处理不仅消除了数据集中可能存在的缺失字段、重复字段，还根据本研究课题修改了数据形式，并降低数据维度，减少运算量。

# 基于卷积神经网络的评分模型

## 实验准备

### 软件依赖

表4-1



### 评价标准

人们总会期望训练出来的模型能够尽可能地准确。换言之，我们要求模型对输出值的预测误差（error）尽可能地小。所以如果我们假定输入变量*X*与输出变量*Y*之间存在函数关系

(4-1)

其中是服从(0, )正态分布的不可避免的噪音变量，则对于*f(X)*的一个拟合模型，预测模型在点*x*处的误差的平方期望值为

(4-2)

即

(4-3)

式（4-3）当中右侧三项依次为偏差、方差与噪声，即预测误差可以认为是由这三项组合而成。偏差刻画的是学习模型本身的拟合能力，其值越小，说明拟合程度越高；方差刻画的是拟合模型的稳定性，通常认为，模型在检测集当中方差越小，则模型稳定程度越高；噪声表示的是存在于变量之间的真实函数关系却无法消除的不可知量，它刻画了模型学习的难度。

基于这一假设，模型应要求尽可能降低模型拟合当中的偏差与方差。而针对本次实验数据，本实验将采用均方根误差与误差标准差两指标对模型进行评价。

均‌方‌根误‌差（Root Mean Square Error, RMSE）的公式为：

(4-4)

其中，分别表示第*k*个数据的预测值和真值。RMSE度量的是数据整体偏差情况；它作为衡量模型拟合偏离程度的指标的优势在于，相比于均方误差（Mean Square Error, MSE）等指标，RMSE没有改变变量的量纲，与之后的数据归一化等去量纲化操作保持一致。

误差标准差（Standard Variance）则为模型预测值与实际值之间的差值的标准差，它刻画的是数据扰动的幅度。同样地，误差标准差也保持了数据量纲的一致性。

## 训练集与测试集数据的准备

由于本实验所采用的数据集含有多个指标，各指标的含义、特性各不相同，所以它们可能具有不一样的数量级。所以如果各指标之间的数量级相差特别大时，机器学习过程中数值高的指标就会比数值低的贡献更显著。所以，对数据分析结果的可靠性的追求促使了数据无量纲化的需求，即本数据集有必要进行数据标准化。另外，数据标准化还有提升模型的收敛速率和精度等优点。

本环节的数据标准化将采用离差标准化法，即如下所示

(4-5)

其中和分别表示数据样本中的最大值以及最小值。离差标准化相对其他标准化手段的优势在于其作为线性方法，计算简单；而且通用性较强，因为无需考虑原本指标数据的分布情况。经过离差标准化后的数据，由于所有样本点均归整至[0,1]之内，且区间[0,1]中任意相同长度的间隔

另外，出于实验训练效果的需要，数据集将会被划分训练集与测试集两类。不过，在实际实验中通常建立在训练集上的模型尽管效果不错，但是对测试集的效果则没有那么令人满意了。这根本原因在于数据划分不均匀而导致的模型欠拟合现象。为了规避欠拟合的发生，本实验在训练集还设有“建模集”与“预报集”，即在实验当中只采用大部分训练集数据进行建模，留下少部分数据样本作为预报集以减少拟合误差。

基于上述规避误差的思路，本实验将采用5折交叉验证法进行模型训练，即将训练集均分成5等分，保留其中一份数据用做验证，其余样本作训练；因每子集分别验证一次，则总共进行5回合验证。

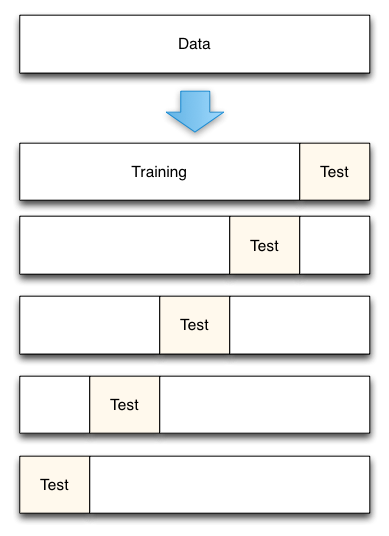


图4-1 K折交叉验证数据划分示意图（K=5）

## 卷积神经网络的构建

从TensorFlow对数据类型的定义来看，本次研究所用的数据均为一阶张量（tensor），而卷积神经网络更适用于图像等二阶张量。针对这一点，本模型构建的主要思路是将一阶张量转化为二阶以便网络学习。有关本卷积神经网络的关键参数如表所示：

表4-2 CNN模型参数表



其中，值得注意的是，之所以输入数据会有16个指标维度，是因为降维后的数据的综合指标数为9（33），而3是奇数，不利于池化层对图像的压缩操作，于是在每条数据后方插入全零矩阵将其补至16（44）维。其代码实现方式如下：

xs = tf.placeholder(tf.float32, [None, 16])

x\_image = tf.reshape(xs, [-1, 4, 4, 1])

本实验中的卷积层依次包含32、64个特征平面，是为了提升模型对数据内部模式的学习效果。而在积神经网络的隐含层中，除了池化层只需调用TensorFlow中的池化方法（本文采用最大池化）以外，卷积层与全连接层均基于这一朴素回归思路集成，其伪代码形式如下：

W = weight\_variable(patch\_size, input\_size,output\_size)  # 权重函数

b = bias\_variable(output\_size)  # 偏置量函数

h = activate(integrate(input\_x, W) + b)

# 激活函数，integrate()函数在卷积层与全连接层分别指代卷积操作（con2d）和矩阵乘积（matmul）

最后，损失函数的实现如下：

cross\_entropy = tf.reduce\_mean(tf.reduce\_sum(tf.square(ys - prediction), reduction\_indices=[1]))

## 卷积神经网络模型的优化

模型的优化环节主要着重关注神经网络的激活函数（Activation Function, AF）的选用。这是因为目前我们无法得知哪一种激活函数的效果更佳，所以需要通过实验比较采用不同激活函数时模型的拟合来检验各种激活函数的效果。本次激活函数将验证RELU、Softplus、ELU、Sigmoid、Tanh、Softsign这6个激活函数在卷积神经网络模型中的应用效果。

由于TensorFlow均已定义了上述6种激活函数的封装方法，所以本环节可以直接调用各自对应的方法即可。而由于本处采用的是5折交叉验证法，所以本处我们将比较的是模型在分别应用6种不同激活方法时，建模集与验证集各自分别的RMSE。另外，为了确保实验结果的精准度，本实验对六种激活方法分别作了3次检验，换言之，本实验可称作35折交叉验证。

## 实验结果与分析

表4-3 CNN模型采用各激活函数时的拟合程度



根据表4-3，除了RELU以外（每次检验下的RMSE差别较大），其他函数在建模集下的效果都较为稳定。进一步考察发现，采用Softsign和ELU时的建模集RMSE在每一次检验都是相对较小的两个，其中Softsign的数值比ELU的更小。在检验集中采用不同激活函数的情况也大致与建模集的情况一样，但是三次检验却反映出采用Softsign作为激活函数对未知数据的拟合程度不如采用ELU时的高。这反映了以Softsign函数建立的卷积网络模型可能“适应”了建模集数据，对数据集以外的样本泛化性能不强，简而言之，该模型相对于以ELU建立的模型的过拟合程度略高。

表4-4



其次，从误差标准差的表现情况来看，采用ELU作为激活函数时的数值是最小的，说明此时的模型相较于其他模型更不易受到新数据加入所产生的扰动的影响，这印证了上文对使用Softsign可能导致过拟合的猜测。因此，综合两方面的考虑，确定优化的卷积神经网络模型的激活函数为ELU。

最后一步便是考虑优化过后的模型在测试集当中的表现。本人利用其中一个ELU函数作为激活方法的卷积神经网络模型对测试集进行检验，此时训练集与测试集下的RMSE分别为0.11203452、0.19352598，误差标准差分别为0.102368313，0.18246357。模型在测试集当中的拟合程度与训练集基本一致，并且稳定程度不易受数据影响，足以说明目前的模型已具备较好的泛化能力。

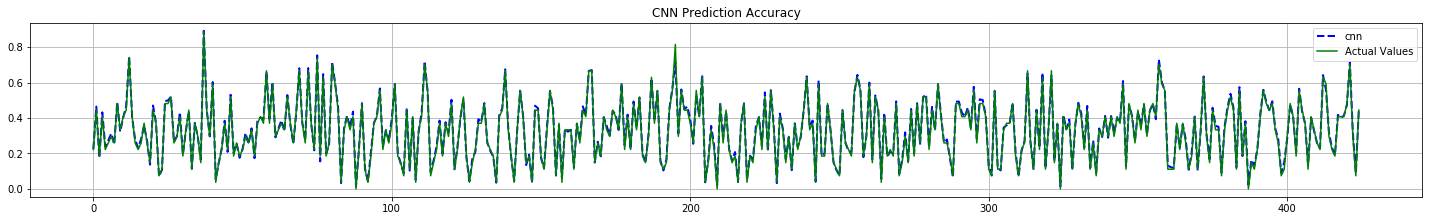


图4-1 卷积神经网络模型预测值与真值比对示意图

## 小结

本章主要讲述的是用于球员综合评分的卷积神经网络模型的构建过程。首先介绍的是关于这次实验的准备工作，包括所依赖的软件环境以及用于考量模型好坏程度的指标。本实验利用能支持实时代码的Jupyter Notebook完成，并且全程均在Windows 10操作系统环境实现，涉及Python、TensorFlow、sklearn、numpy、pandas等工具。

在数据准备环节有两点需要注意，首先本网络将采用离差归一化的方法对各指标数据进行标准化，使各自满足均匀分布；其次，为了提升模型准确程度，降低过拟合程度，本实验将采取35折交叉验证法，保证数据集能够被充分学习。紧接着介绍了搭建卷积神经网络思路，即把原为一阶张量的原始数据转换成二阶张量，以方便利用TensorFlow工具包的方法。

在完成拓扑结构的实现以后，由于目前无法得知各激活函数在模型中的适用程度，本实验的优化过程着眼于激活函数的选择上。实验比较了分别采用RELU、Softplus、ELU、Sigmoid、Tanh、Softsign这六种激活函数各自的拟合效果，从实验结果看到，ELU作为激活函数时效果最佳，于是将其作为优化模型的激活函数。

最后一步考虑模型在测试集当中的拟合效果。根据预测模型的可视化效果，预测值已达到与真实值走势一致的水平，而且在偏差与误差方差这两评价指标上模型的表现同样让人满意，说明此时的模型已经具备相当足够的泛化能力。

# 基于LSTM的评分模型

## 实验环境

表5-1 实验所需软件准备工作



## LSTM模型的构建

本实验数据的主要准备工作与4.2介绍的一致。本模型的关键参数如下所示：

表5-2 LSTM模型参数表



首先我们考虑的是隐含层拓扑结构的实现。本神经网络包括3个隐含层结构，其中仅有中间一层具备学习能力（即神经元层），其余两层网络的功能在于转换数据张量的阶与维度。以下为神经元层的实现过程：

**def** add\_cell(self):

lstm\_cell = tf.nn.rnn\_cell.BasicLSTMCell(self.cell\_size, forget\_bias=1.0, state\_is\_tuple=True)

        # 数据初始状态

        with tf.name\_scope('initial\_state'):

            self.cell\_init\_state = lstm\_cell.zero\_state(self.batch\_size, \

dtype=tf.float32)

        # 非初始状态的数据，time\_major=False 表示时间主线不是第一列batch

        self.cell\_outputs, self.cell\_final\_state = tf.nn.dynamic\_rnn(

            lstm\_cell, self.l\_in\_y, initial\_state=self.cell\_init\_state, \

time\_major=False)

需要注意的是，BasicLSTMCell方法中还有activation这一用作指定激活函数类型的参数，这一操作将在下一节进行讲述。

考虑到LSTM对时序数据的学习能力，本实验的数据处理思想是将所有数据按一定时间顺序导入，即将普通数据转换作时序数据。具体来讲，就是把数据（训练集）排放在一时间轴上，按顺序将样本规模固定的数据分批次喂给模型学习：

**def** get\_batch\_data():

**global** train\_x, train\_y,BATCH\_START, TIME\_STEPS

    x\_part1 = train\_x[BATCH\_START : BATCH\_START+TIME\_STEPS\*BATCH\_SIZE]

    y\_part1 = train\_y[BATCH\_START : BATCH\_START+TIME\_STEPS\*BATCH\_SIZE]

**print**('时间段=', BATCH\_START, BATCH\_START + TIME\_STEPS \* BATCH\_SIZE)

    seq =x\_part1.reshape((BATCH\_SIZE, TIME\_STEPS ,INPUT\_SIZE))

    res =y\_part1.reshape((BATCH\_SIZE, TIME\_STEPS ,1))

    BATCH\_START += TIME\_STEPS

**return** [seq , res  ]

另一方面，为了确保所有数据都能够被模型充分学习，即模型需要录入所有数据。为了达到这一效果，鉴于数据量达到2122，剔除20%的数据样本作为预测集，并采取5折交叉检验法后，应规定本实验中时间段应为125方能保证数据能被最大程度学习。所以拟合模型的运行代码如下所示：

**for** j **in** range(200):  # 训练200次

        pred\_res=None

**for** i **in** range(125):  # 把整个数据分为125个时间段

            seq, res = get\_batch\_boston()

**if** i == 0:

                feed\_dict = {

                        model.xs: seq,

                        model.ys: res,

                        # create initial state

                }

**else**:

                feed\_dict = {

                    model.xs: seq,

                    model.ys: res,

                    model.cell\_init\_state: state

# 令上一次运行状态作为本次运行得初始状态

                }

            \_, cost, state, pred = sess.run(

                [model.train\_op, model.cost, model.cell\_final\_state, model.pred],

                feed\_dict=feed\_dict)

            pred\_res=pred

            result = sess.run(merged, feed\_dict)

            writer.add\_summary(result, i)

**print**('第{0}次 学习损失函数值 ：'.format(j + 1), round(cost, 8))

        BATCH\_START = 0

## LSTM模型优化实验结果与分析

与4.4节中有关卷积神经网络的优化一样，本节关于LSTM模型的优‌化‌同‌样关注的‌是模‌型中激活‌函‌数‌的选择。

采用不同激活函数是模型的拟合效果如下所示：

表5-3 LSTM模型采用各激活函数的拟合程度



表5-3的结果显示，无论采用哪一种激活函数，LSTM模型的损失值，即在建模集中的RMSE数值均小不超过0.04，并且三次检验的结果甚至只为千分位级别，说明LSTM作为足球运动员的综合评分模型，对激活函数的选择相对不敏感。另一方面，如表5-4所示，误差偏差的表现也从数据扰动的层面说明模型对激活函数的不敏感性。

表5-4 LSTM模型采用各激活函数时的数据扰动程度



鉴于模型的不敏感性，我们很难从三次不同的实验数据当中找到很好地比较法则。所以出于方便，本最优激活函数的选择环节改变了思路：比较三次实验中各不同激活函数作用的RMSE与误差标准差的均值，如表5-5所示。

表5-5 LSTM模型各激活函数时的偏差程度



可以看到，当使用Sigmoid和ELU作为LSTM模型的激活函数时在训练集中都能够有不错拟合表现（Error值较小），而Sigmoid在验证集当中的拟合效果则更胜一筹。因此，规定优化的LSTM以Sigmoid函数作为激活函数。

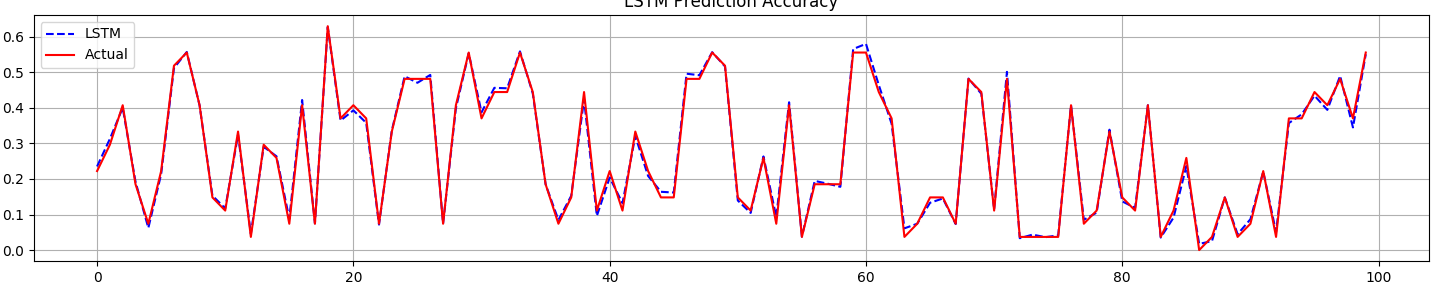


图5-1 LSTM模型预测值与真值比对示意图

最后需要考察的是测试集上模型的拟合情况。本人利用其中一个Sigmoid函数作为激活方法的卷积神经网络模型对测试集进行检验，此时训练集与测试集下的RMSE分别为0.03543196、0.10527381，误差标准差分别为0.03490613，0.08243576。模型在测试集当中的拟合程度与训练集基本一致，并且稳定程度不易受数据影响，足以说明目前的模型已具备较好的泛化能力。

## 小结

本章介绍的是基于LSTM的球员综合评分模型的构建与实验结果检验本章重点是LSTM循环神经网络的构建。本网络模型的构建思路是将原本没有时序关系的数据按照向模型导入，由程序根据数据之间的时间紧密程度决定数据的取舍。至于数据取舍的实现，根据LSTM的理论知识，该网络模型因为引入了门机制而拥有了记忆，不过在借助TensorFlow工具包的情况下，本实验仅需要调用LSTMBasicCell这一类对象，设置遗忘偏差等基本参数即可基本实现网络的隐含层结构。除此以外，与第四章对于卷积神经网络模型的优化过程一致，本章同样根据采用不同激活函数时的模型拟合效果进行优化改良。实践证明，本次研究需要探讨的六种激活函数在应用当中的拟合效果都已经十分显著，且LSTM模型对激活函数有着不敏感的特性。所以为了选择最佳的激活函数，此环节本人改变研究思路，转而比较三次检验下采用各激活函数的模型的平均偏差。经比较，尽管采用ELU的模型在建模集中的预测较少出错，但是在检验集中的情况则比不上采用Sigmoid函数的模型。另外，采用Sigmoid在LSTM模型的激活函数在建模集当中的表现仅次于ELU。因此，本次选择Sigmoid作为优参函数。

最后考察模型在测试集当中的表现，利用其中一个Sigmoid函数作为激活方法的卷积神经网络模型对测试集进行检验，RMSE与误差的标准差，以及预测数据的可视化效果都表明目前的LSTM模型已经具备较高的泛化能力。

本实验证明了LSTM模型在本球员表现综合评分这一情景下已经具备了可靠性与稳定性。结合第四章卷积神经网络模型的表现，再一次证实了深度学习在足球运动员综合表现评分这一情景中的应用的可行性。

# 总结与展望

## 本文主要工作与结论

本文围绕“足球运动员综合评分”这一主题，主要从数据获取与评分模型构建两方面展开深入探讨。

首先，本文介绍了本课题的背景，包括针对综合评价模型的研究、深度学习在体育界的应用以及深度学习在足球的应用。我们了解到目前足球与机器学习等信息技术的联系已经相当紧密，然而由于机器学习要求大规模的特征工程这一弊端导致有关足球运动员综合表现评价这一主题的研究仍有十分广阔的发展空间。

然后，针对上述球员综合评价当前存在的弊端，本文介绍了自组织、自学习、自适应能力更强的深度学习与本课题研究结合的可能，并为此展开实验介绍陈述。

随后，在实验介绍开始前我们了解了有关本次实验的知识技术基础，包括网络爬虫、Scrapy框架、HTML、主成分分析、卷积神经网络、循环神经网络、LSTM、因子分析以及多类常见激活函数。

至于实验的第一步——数据获取，本文阐述了网络爬虫技术实现中的一些难点与痛点，比如Scrapy框架的搭建，数据抓取规则和反爬机制的应对方法，以及采取爬虫提取数据的原因：目前很难获得免费、公开的足球赛事数据源，而网络爬虫便是一个批量获取信息的有效途径。

另外，在建模开始之前，出于方便后续任务、避免过拟合、降低计算量的考虑，本文首先对数据集进行了预处理，处理工作包括修复缺失值、剔除重复数据和降维。其中降维采用的主成分分析与因子旋转相结合的方法。

最后，本文分别利用卷积神经网络以及LSTM搭建球员综合评分模型。需要注意的是，本次实验主要强调有监督学习模式，即假定第一步骤中获取得到的球员综合评分数据具备足够的权威性，可作为模型输出的期望值。本文具体讲解了两个网络结构实现所需要进行的准备工作，包括软件环境与数据准备，和模型的构建主要思路，以及最终各自的运行结果。卷积神经网络模型主要要求将原始数据转换成类似于图像的二阶张量，从而更加方便利用TensorFlow方法；同样地，考虑到LSTM主要针对时序问题而提出，本LSTM模型同样需要数据转换作时序形式，即将数据分批次导入模型，程序按照数据之间的时间紧密程度决定数据取舍。另外，尽管研究已经成功构建两个网络模型的拓扑结构，但是由于目前学术界对于各激活函数的优劣程度的评价尚无定论，本次模型构建第二个环节对6种（RELU、Softplus、ELU、Sigmoid、Tanh、Softsign）常用的激活函数在CNN和LSTM模型的分别应用中的效果进行了比较，并选择出对两个模型适用程度的函数作为优化模型的激活函数。根据比较结果，CNN模型选择了ELU作为最优激活函数，而LSTM的最有激活函数则为Sigmoid。

根据第四章与第五章的结果，无论是基于卷积神经网络，抑或LSTM，深度网络线网络学习模型在球员综合评分的应用都具备了相当强的可行性；二者都具备根据少量样本数据发掘全体或大量数据当中内在规律的能力。但是相对而言，如果仅根据实验数据集的拟合效果，LSTM是更胜一筹的一方。不过，由于LSTM因为参数设置的原因将数据之间的时序关系纳入了考虑，然而实际上本实验所采纳的数据除了同一球员不同赛季的表现以外并没有十分显著的时间层面上的相关性，这表明了尽管LSTM模型表现出色，但是对问题的解释能力未必足够强大。因此，本人更倾向于选择较为直观的卷积神经网络作为评分系统的模型。

总而言之，深度学习在球员评分领域是可行的，但不同模型各有自己的优劣。考虑到球员评分在转会运作、战术制定等各种不同方面均有着广阔应用前景，我相信深度学习将会为球员评分这一研究课题带来更多创新与优化改进。

## 实验中存在的不足与对未来研究的展望

随着信息技术的发展，足球甚至体育与数据科学、信息科技将会越来越紧密地联系在一起。人们总会期望用更出色的算法去挖掘出数据内潜在的规律。所以我认为，如何从其他领域借鉴与运用先进的方法思路，结合足球运动的特点进行优化改良，以此来满足人们在足球管理、博彩竞技等方面的需求，是接下来的研究工作重点。本次研究主要基于前人的研究经验与贡献而完成，然而由于笔者能力欠缺，加上精力有限，研究工作不可避免地会存在瑕疵。而下面三点是本人认为需要且能够改进的方面：

1. 数据量规模不够大。根据创冰信息科技有限公司的官网，他们目前已有实力在每场比赛进行期间实时产生约1500条数据项目，而在本次实验中只能够在其网页中爬取得到最多100左右的项目指标。显然，该网页由于保护企业自身的“数据资产”而并未将其发布。所以，理论上假如能够获取得到上述受保护的数据项目，这将大大丰富数据集的特征，相信对本次实验的效果将大有提升。另外，创冰DATA成立于2014年，并自创立起便成为了中超赛事的数据供应商，所以本次网络爬虫操作最多只能获取2014赛季至今的中超比赛。如果能够通过其他渠道能够获得再往前的数据的话，比如联系2014年以前的中超联赛数据供应商，我相信这会更能体现深度学习处理大数据情形的先进性。
2. 各模型中的参数，如网络层数、迭代次数等，没有经过优化，因此模型效果仍有提升空间。
3. 网络结构的调整。由于影响一位球员的综合表现的因素是多方面的，而且如前文所言，大量潜在有关的因素数据无法获取，所以当这些数据有机会获取得到时，我们便需要采取新的方式去提取数据中内在的联系性。在这种情况下，其他模型或者各种模型的组合形式或许能够发挥更好的拟合效果。
4. 以创冰公司评定的比赛评分的赛季均值作为输出期望值的做法值得商榷。由于球员的比赛评分逻辑因商业机密等原因未能公开，我们无法得知该评分的准确程度与可信程度。考虑到目前创冰作为国内最出色的体育数据生产、分析商的地位，我认为该评分仍然具备相当高的可参考价值。

参考文献

1. FIFA. 2018 FIFA World Cup Russia: Global Broadcast and Audience Executive Summary[R/OL]. https://resources.fifa.com/image/upload/2018-fifa-world-cup-russia-global-broadcast-and-audience-executive-summary.pdf?cloudid=njqsntrvdvqv8ho1dag5, 2018-12-21.
2. Frank Keogh, Gary Rose. Football betting - the global gambling industry worth billions[N/OL]. https://www.bbc.com/sport/football/24354124, 2013-10-03.
3. K. Fukushima. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,[J]" Biol. Cybern., 36, 193–202, 1980.
4. LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436.
5. Rusty Simmons. Competitive fire helps Kirk Lacob make his own name with Warriors[N]. SFGATE, 2015-06-20.
6. 李红, 朱建平. 综合评价方法研究进展评述[J]. 统计与决策, 2012, 9: 7-I1.
7. Sun W, Niu D X, Shen H Y. Comprehensive evaluation of power plants' competition ability with BP neural networks method[C]//2005 International Conference on Machine Learning and Cybernetics. IEEE, 2005, 8: 4641-4644.
8. Sun W, Shen H, Yang C. Comprehensive Evaluation of Power Plants' Competition Ability with SVM Method[C]//2006 International Conference on Machine Learning and Cybernetics. IEEE, 2006: 3568-3572.
9. Miller A, Bornn L, Adams R, et al. Factorized point process intensities: A spatial analysis of professional basketball[C]//International Conference on Machine Learning. 2014: 235-243.
10. Hamilton M, Hoang P, Layne L, et al. Applying Machine Learning Techniques to Baseball Pitch Prediction[C]//ICPRAM. 2014: 520-527.
11. Sinha S, Dyer C, Gimpel K, et al. Predicting the NFL using Twitter[J]. arXiv preprint arXiv:1310.6998, 2013.
12. Hervert-Escobar L, Hernandez-Gress N, Matis T I. Bayesian based approach learning for outcome prediction of soccer matches[C]//International Conference on Computational Science. Springer, Cham, 2018: 269-279.
13. 夏飞. 基于遗传算法及BP神经网络的足球比赛结果预测研究[D].重庆师范大学, 2017.
14. 吴金龙. 基于决策树C4.5算法的足球赛事预测[D].广东工业大学, 2016.
15. Baboota R, Kaur H. Predictive analysis and modelling football results using machine learning approach for English Premier League[J]. International Journal of Forecasting, 2018.
16. 孙立伟, 何国辉, 吴礼发. 网络爬虫技术的研究[J]. 电脑知识与技术, 2010, 6(15): 4112r4115.
17. LeCun Y, Bottou L,Bengio Y,et al.Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
18. Tamir M, Oz G. Real-Time Objects Tracking and Motion Capture in Sports Events: U.S. Patent Application 11/909,080[P]. 2008-8-14.
19. Caporale, T. & Collier, T.C. To Three or Not to Three?[J]. 1936-4768/ 0195-3613. 2014-10-29.
20. Ford R A. Trade secrets and information security in the age of sports analytics[M]//The Oxford Handbook of American Sports Law. 2018.
21. Schumaker R P, Jarmoszko A T, Labedz Jr C S. Predicting wins and spread in the Premier League using a sentiment analysis of twitter[J]. Decision Support Systems, 2016, 88: 76-84.
22. Kose H, Akin H L. Object recognition in robot football using a one dimensional image[C]//The Tenth Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN 2001). 2001: 291-300.
23. Shi L I, Jiang C, Zengqi S U N. Structural design and implementation of Tsinghua Robot Soccer Team [J][J]. Journal of Tsinghua University (Science and Technology), 2001, 7.
24. Rumelhart D E, Hinton G E, Williams R J. Learning internal representations by error propagation[R]. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

致谢

行文至此，意味着笔者本科生涯即将画上句号。从懵懂而充满好奇的大一新生走来，我在求学路上有幸找到自己真正感兴趣的研究方向——数据分析，并在此深耕。更让我觉得幸运的是，我四年本科求学路上收到来自各方的关心与支持，在此本文特地向他们表示诚挚的谢意。

首先，我要感谢的是我的父母与其他家人。正是他们含辛茹苦的工作，才能让我能够安心学习。另外，我还想借此机会对爸爸妈妈说声抱歉，孩儿身在远方，难以顾及家长里短，实属无奈；在你们即将安享天伦之乐的日子还需劳烦两位照顾长辈，我更是倍感心疼。如今孩儿毕业在即，你们大可以筹划自己的退休计划，生活起居之事交由孩儿打理即可。

其次，我需要感谢的是各位曾担任笔者的任课教师。华园实属教书育人之宝地，每位老师都各具教学特色，对华园学子充满关怀。其中，我尤其需要感谢的是本人的毕业论文指导老师凌黎明教授，他儒雅随和的性格给了我广泛拓展研究范围的空间。而另一方面，凌老师又是一位治学态度严谨的研究者，他深厚的理论知识与认真负责的作风深刻地影响了我，不断督促我朝着既定方向前行。因此，我在此祝愿包括凌黎明教授在内的所有执教过的老师，事业顺利，生活美满，身体健康！

最后，我还想感谢在广州绿网实习期间的领导以及各位同事。他们得知并了解本人工作期间同时还有毕业论文这一重要任务，都对我给予了莫大的支持与鼓励。其中，我想感谢的是我的直属领导“欣哥”罗欣。欣哥作为本文本公益组织的爬虫项目牵头人，为我传授了不少有关网络爬虫的知识经验，对本人毕业设计的工作大有帮助。借此机会，我祝愿广州绿网能够越办越好，早日实现组织目标！

蒙安哲

2019年5月8日星期三