浙江大学

硕 士 学 位 论 文

论文题目：基于卷积神经网络的电商数据深度挖掘

作 者 姓 名：钊魁

指 导 教 师：王灿副教授

学科(专业)：计算机科学与技术

所 在 学 院：计算机科学与技术学院

提 交 日 期：二〇一七年一月

A Dissertation Submitted to Zhejiang University
for the Degree of Master of Computer Science

TITLE: Deep Data Mining in E-commerce Based on Convolutional Neural Networks

Author: Kui Zhao

Supervisor: Assoc.Prof. Can Wang

Subject: Computer Science and Technology

College: College of Computer Science

Submitted Date: January, 2017

摘要

近年来，快速发展的电子商务为人们带来了极大的便利。电商所处的商业环境相较传统商业环境具有更强的动态性与复杂性，这带来了诸多挑战，而数据挖掘技术可以帮助人们更好地应对这些挑战。传统数据挖掘技术无法有效地利用电商中的海量数据，它依赖于耗时、耗力的特征工程，得到的模型可扩展性差。深度学习技术可以有效地利用大量数据，且可以实现自动化地从原始数据中抽取有效特征，具有更高的可用性。在本文中，我们利用深度学习中的卷积神经网络对电商数据进行挖掘，针对商品搭配推荐与商品销量预测这两个方面，设计了一系列有效的算法及优化方法。具体而言，本文的主要研究内容包括：

首先，商品搭配具有广泛应用，如基于用户已购买商品向其推荐可能购买的商品。传统方法通过分析商品历史共同购买记录生成搭配信息，它无法为没有历史购买记录的新商品生成搭配信息。在本文中，我们观察到商家会把商品所有重要属性信息放在标题中，于是设计了一个对拍卷积神经网络对两个商品标题组成的短文本对建模，将文本信息从原始的符号空间映射到特定的样式空间，进而在样式空间中计算两个商品间的搭配程度。其次，商品销量预测对商业决策至关重要，它有助于商家对人力、物力与仓储等诸多方面做出更优的管理。基于时序分析的方法仅能对那些销量变化规律明显的商品做出准确预测；虽然传统机器学习方法可以通过特征工程来考虑更多信息，进而取得更高的准确性，但特征工程限制了模型的可扩展性。在本文中，我们设计了一个新颖的模型，它可以从原始结构化时序数据中通过卷积神经网络自动化提取有效特征，并进一步利用这些特征实现商品销量预测。最后，在真实电商数据集上验证了我们提出算法的有效性。

关键词：深度学习，卷积神经网络，电子商务，数据挖掘，商品搭配，自然语言处理，样式匹配，销量预测，特征学习，时序分析

Abstract

In recent years, the E-commerce develops rapidly and has brought great convenience for people. While dynamic and complex business environment in E-commerce raises great challenges, data mining techniques help to overcome these challenges. However, traditional data mining techniques cannot effectively utilize the massive amount of available data in E-commerce and the models constructed by them lack of practicability. The reason is that they rely on the manual feature engineering, which is usually a difficult, time-consuming task and requires expert knowledge. Deep learning models can make full use of these data by extracting effective features automatically from the raw data and thus have a higher practicability. In this paper, focusing on the style match and sales forecast, we design a series of efficient data mining approaches in E-commerce scenarios based on Convolutional Neural Networks. Specifically, the principal studies of this paper are listed as follows:

Firstly, style match can be exploited in many commercial applications, such as recommending items to users based on what they have already bought. The traditional frequent item-set mining methods generate match items by analyzing the historical purchasing patterns and cannot handle new products without historical records. Based on the observation that online sellers will place most of the important attributes of a product in its title description, we design a Siamese Convolutional Neural Network in this paper and feed it with title pairs of items. Those short text pairs will be mapped from the original space of symbolic words into some embedded style space, where the compatibility between two items is calculated. Secondly, sales forecast has a crucial impact on making informed business decision and can help us to manage the workforce, cash flow and resources etc. Traditional sales forecast is based on time series analysis of historical

sales and can only handle well the commodities with stable or seasonal sales trend. Though more recent learning-based methods improve the forecast accuracy by capturing more information in the models, they require case-by-case manual feature engineering and thus are limited in their applicability. In this paper, we design a novel approach to learn effective features automatically from the structured time series data using the Convolutional Neural Network. When fed with raw log data, our approach can automatically extract effective features from that and then forecast sales using those extracted features. Finally, we test our approaches on several large real-world datasets in E-commerce and the experimental results validate the effectiveness of our approaches.

Keywords: Deep learning; Convolutional Neural Networks; E-commerce; Data mining; Complementary Recommendation; Natual language processing; Style match; Sales forecast; Feature learning; Time series

目录

插图

表格

第 1 章 绪论

1.1 引言

近些年来，随着互联网的普及，电子商务快速发展，为人们的生活带来了诸多便利。与传统商业相比，电商中的产品数量巨大、更新周期短，更为动态而复杂的商业环境在多个方面带来了巨大挑战。数据挖掘技术可以帮助人们更好地应对这些挑战，如从海量商品中找出相搭配的商品来进行推荐与预测商品销量来辅助商业决策等。然而，传统的数据挖掘技术依赖于人工特征工程（feature engineering)，它不但费时、费力，还需要做特征工程的人员拥有特定领域的专业知识[1]。这限制了传统数据挖掘技术所得到模型的可扩展性及可用性，使之无法有效地利用电商中大量可用数据。

与传统技术不同，深度学习（deep learning）技术可以自动化地从大量原始数据中抽取有效特征，因此通过深度学习所建立的模型有更强的可用性[2]。深度学习又被称为神经网络（neural networks），其灵感来源于神经系统：将神经元抽象为结点，突触抽象为边，结点通过边连接在一起。神经网络通过介于输入层与输出层之间的连接关系刻画函数，边上的权重便是所刻画函数的参数。在输入层与输出层之间的隐含层中，特征被自动化地从原始数据中提取出来，然后应用于后续输出层的分类或拟合问题中。卷积神经网络（convolutional neural networks）是深度学习中最为重要的框架之一，它可以很好地利用数据中的“时空局部性”这一先验信息[3]。

在本文中，我们利用卷积神经网络在电商数据挖掘中做出了一些全新的尝试，主要包括商品搭配推荐与商品销量预测两个方面：商品搭配推荐具有广泛应用，如基于用户已购买商品向其推荐可能购买的商品；而商品销量预测对商业决策至关重要，它有助于商家对人力、物力与仓储等诸多方面做出更优的管理。

1.2 研究背景

商品搭配推荐与商品销量预测一直备受人们关注，从最开始完全由人工手动完成，到后来借助于计算机等工具半自动化完成，再到近些年来借助于智能技术完全自动化完成，其所涉及到的算法在不断地演化。

1.2.1 商品搭配推荐

商品搭配信息在很多商业场景中都有应用，比如根据用户已经购买的商品向其推荐可能购买的商品，或直接向用户推荐商品购买套装等。生成这些信息最直接的做法是人工标定：让若干搭配达人来手工标定哪些商品是相搭配的。在传统商业中商品数量较少，这一方法具有较高的可行性；而电商中商品数量巨大，虽然可以通过众包（crowdsourcing）等方式实现大规模标注，但成本会非常高。如何自动化地生成商品间搭配信息，无论是对学术界还是工业界，一直都是个备受关注的问题。

自动化生成商品搭配信息的最经典算法是“频繁项集挖掘”（frequent item-set mining)[4]，它从商品的历史购买记录中挖掘出商品间的搭配关系，其基本假设是经常被共同购买的商品往往是相搭配的，比如最经典的案例：“啤酒”与“尿布”。然而频繁项集挖掘算法有一个致命的缺点，那就是冷启动（cold-start）问题，即无法为没有历史购买记录的新商品生成搭配信息[5]。而电商中的商品更新周期短，很大一部分商品都是没有历史购买信息的新商品。

基于内容的商品搭配推荐算法直接利用商品的描述信息进行搭配，因为电商中的所有商品都有对应的描述信息，所以这些方法从根本上克服了“冷启动”问题。如 McAuley 等人[6] 利用商品的描述图片计算商品间的搭配程度，他们以两个商品描述图片间的相似性来衡量两个商品的搭配程度。虽然描述图片间的相似性在一定程度上可以衡量两个商品的搭配程度（尤其是“服饰”类别），但是相似并不完全等同于相搭配；另外，处理图片所需的计算量大，且图片经常包含大量噪声，这些因素都影响了基于描述图片的算法的实用性。

1.2.2 商品销量预测

商品销量预测在电子商务中是一个非常重要的任务，它有助于商家有效地管理人力、物力与仓储等。销量预测的价值取决于其准确性，过高估计销量会导致商品积压，过低估计销量会导致商品脱销。在传统商务中，涉及到的商品数量少，销量预测直接由人来完成。较大的商家会利用信息系统对历史销量数据进行保存，然后利用历史销量数据辅助人做出更为准确的预测。如果涉及到的商品数量较多，可以利用时序数据分析（time series analysis）技术自动化地对每个商品的销量做出预测[7]。人工预测销量只适用于商品数量较少的场景，利用时序数据分析技术预测销量只适用于销量稳定或规律明显的商品[8]。电商中商品数量众多，商业环境相较传统商业更为动态而复杂，上述方法都无法有效地应对。

虽然电商中商品销量预测面临着更大的挑战，但与传统商业不同的是大量数据可以被轻松地收集起来并加以利用，如浏览次数（PV）、浏览人数（UV）与价格（PAY）等，利用这些数据能够有效地提升销量预测的准确性。通常的做法是通过监督学习（supervised learning）对这些数据加以利用，更具体地说是回归模型（regression model）。首先，通过人工特征工程从已有可用数据中提取若干特征；然后，将提取的特征作为线性回归（linear regression）或 GBRT（Gradient Boosting Regression Tree）等回归模型的输入，将销量作为回归模型的输出；最后，在训练数据（training dataset）上训练销量预测模型，然后对新的数据点做出预测。

传统监督学习方法的准确性依赖于人工特征工程所提取特征的质与量，而提取有效特征是个费时、费力的任务，且需要做特征工程的人员拥有特定领域的专业知识。另外，特征工程需要针对特定的应用进行，也就是说当处理新数据或新任务时，要再次实施特征工程。比如，有更多的数据可用于商品销量预测时，需要专业人员对这些新数据做完特征工程之后，预测模型才能够有效地利用这些新数据。可见，特征工程限制了模型的可扩展性，使之无法快速、有效地利用电商中与日俱增的可用数据。

1.2.3 卷积神经网络

特征学习能够从原始输入数据中自动化地提取有效特征，这些特征可以进一步用于特定的任务当中。深度学习又被称为神经网络，它是最常用的特征学习方法之一，其灵感来源于神经系统。最近，很多问题的最优方法都来源于深度学习，而卷积神经网络是深度学习中最为重要的框架之一，它将数据中的“时空局部性”很好地融合到模型当中。

卷积神经网络最早由 LeCun 等人于 1998 年提出[3] 并长期未出现在人们的视线当中。它最近的复兴很大程度上要归功于 AlexNet[9] 在 2012 年 ILSVRC 比赛（ImageNet Large Scale Visual Recognition Challenge）上赢得冠军且误差降低程度与传统方法相比非常可观。之后它席卷了计算机视觉（computer vision）、语音识别（speech recognition）与自然语言处理（natural language processing）等诸多领域的大部分问题，甚至在人类引以为傲的的智力游戏“围棋”上，帮助计算机战胜了世界最强手李世乭[10]。

1.3 研究内容

在本文中，我们利用卷积神经网络对电商数据进行挖掘，针对商品搭配推荐与商品销量预测这两个方面，设计了一系列有效的算法及优化方法。具体而言，本文的主要研究内容包括：

- **商品搭配推荐**。我们观察到商家为了商品更容易被用户通过搜索引擎访问，会把商品所有的重要属性信息放在标题当中。因此商品标题不但包括商品的外观信息，还包括商品的类别、品牌与适合人群等信息。考虑到商品间的搭配本质上是商品属性间的搭配，利用商品标题能够更好地为商品间的搭配关系建模。在本文中，我们设计了一个对拍卷积神经网络（siamese convolutional neural network)，这个对拍神经网络以两个商品标题组成的短文本对作为输入。首先，通过卷积神经网络将每商品的标题表示为一个多

维实数向量；然后，将上述实数向量映射到特定的“样式空间”（style space）中；最后，在样式空间中计算两个商品间的搭配分数。此外，我们还利用最近邻搜索（nearest neighbor search）技术对算法在实际的推荐应用场景中进行加速。在来自淘宝与亚马逊的两个大规模数据集上的验证实验表明，我们所提出的算法能够有效地提升商品搭配生成质量。

- **商品销量预测**。我们提出了一个新颖的方法，利用卷积神经网络从结构化时序数据（structured time series data）中自动化地提取有效特征。首先，我们把与商品相关的原始日志数据转换为特定的“数据框”（data frame）格式，其中原始日志数据包括商品在过去很长一段时间上的销量、价格、浏览次数、浏览人数、搜索次数、搜索人数、收藏人数、加购物车人数等诸多指标；然后，在数据框上应用卷积神经网络提取有效特征；最后，在卷积神经网络的最后一层利用这些特征预测商品的销量。此外，我们还利用样本权重衰减（sample weight decay）与迁移学习（transfer learning）等技术进一步提升商品销量预测准确性。在来自菜鸟网络的大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品销量预测的准确性。

无论是在商品搭配推荐中还是在商品销量预测中，我们所设计的模型都符合端到端（end-to-end）的形式。换句话说，我们的模型以原始数据作为输入、以最终目标作为输出，不需要人工处理数据及干涉模型。

1.4 论文结构

本文共分为五章，图1.1给出了本文的组织结构，其中：

第 1 章为绪论，主要介绍基于卷积神经网络的电商数据深度挖掘的研究背景、关键技术与本文主要研究内容等。

第 2 章为相关工作，主要对电商数据深度挖掘中的商品搭配推荐和商品销量预测两个方面的相关研究工作进行归纳与总结。另外，深入地调研和分析深度学

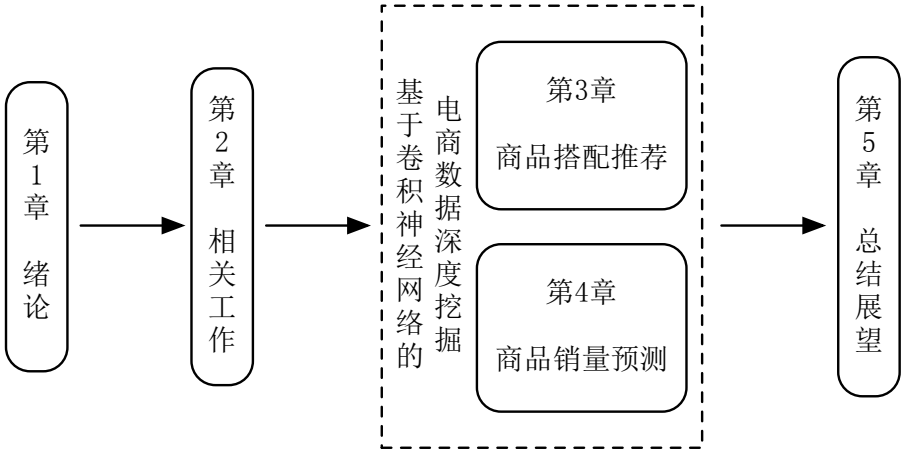


图 1.1 论文组织结构

习中卷积神经网络这一框架的技术内容。在介绍相关工作的同时，讨论现有工作的特点与不足，并简单地介绍本文研究工作如何解决相应问题。

第 3 章为商品搭配推荐，提出基于深度样式匹配的商品搭配推荐算法。首先展示我们如何将商品标题映射为多维实数向量；其次展示如何将上述实数向量映射到样式空间中并计算商品间的搭配分数；再次展示如何在实际的推荐应用场景中加速算法；最后展示在淘宝和亚马逊两个大规模数据集上的验证实验。

第 4 章为商品销量预测，提出基于深度时序分析的商品销量预测算法。首先展示我们如何将原始日志数据转化为特定的数据框；其次展示如何将卷积神经网络应用于上述的数据框并实现销量预测；再次展示如何通过样本权重衰减与迁移学习技术进一步提升销量预测的准确性；最后展示在菜鸟网络大规模数据集上的验证实验。

第 5 章为总结展望，主要对本文的研究内容进行总结并对下一步的工作进行展望。

第 2 章 相关工作

2.1 商品搭配推荐综述

商品搭配推荐一直是个备受人们关注的问题，与其相关的工作最早可以追溯到“频繁项集挖掘”（frequent item-set mining）[4]。这种方法通过分析商品的历史购买记录来生成搭配关系，即经常被共同购买的商品很可能是相搭配的，最经典的案例便是“啤酒”与“尿布”。共同购买关系可以进行传递，比如商品 a 与 b 经常被共同购买，商品 b 与 c 经常被共同购买，那么商品 a 与 c 很可能也是相搭配的。可见，共同购买关系的传递距离与确定搭配所需的共同购买次数是该方法的两个关键超参数（hyper-parameter）。频繁项集挖掘依赖于商品的历史购买记录，所以存在“冷启动”问题：无法为那些没有历史购买记录的新商品生成搭配信息[5]。

为了解决冷启动问题，需要引入除历史记录外的其它信息，比如内容信息（content-based）等。相较历史记录信息，内容信息更为稠密，且不会出现老商品有信息而新商品没有信息的情况，因此从根本上克服了冷启动问题。关于如何利用内容信息解决冷启动问题，更详细的总结可参见 Pazzani 等人的综述[11]。事实上，基于内容信息的方法已有很多商业化应用，比如 Jinni[1] 利用电视节目的内容信息向用户推荐其可能喜欢的电视节目。他们通过人工特征工程的方式从用户评论与节目元数据（包括编剧、导演与演员等信息）中提取特征，然后利用这些特征进一步构建推荐模型。

具体到商品搭配场景，早期基于内容信息的方法大部分聚焦于服饰类商品的搭配上。Vignesh 等人[12] 通过图片分割（image segmentation）算法从街头照片中直接解析出相搭配的衣服，他们的基本假设是街头照片中人们所穿的衣服是相搭配的，因此通过图片分割算法把不同的衣服从图片中分割出来后，这些衣服便形

[1] http://jinni.com

成了具有搭配关系的商品。Yamaguchi 等人[13] 通过计算视觉上的相似，来扩展由图片切割所生成的搭配信息。Di 等人[14] 则做了更进一步的探索，关注于样式相关的精细化属性信息，比如是否有领子、是否为短款与是否有兜等。他们首先人工精细化地标定了若干衣服，形成数据集 WFC，然后利用这一数据集建立起基于属性搭配的衣服搭配模型。Kiapour 等人[15] 通过众包（crowdsourcing）的方式来生成大规模搭配及风格标定数据集。他们设计一款名为“Hipster wars”[2]的在线游戏，利用这款游戏收集了大量的搭配及风格标定数据，并通过实验进一步探索了哪些元素对风格及搭配尤为重要。

上述基于内容信息的方法都是为特定的应用而设计的，其中与搭配相关的工作均针对服饰间相搭配的场景。这些方法或是依赖于人工提取的特征，或是依赖于人工标定的数据集，均无法很好地应用于电商场景下的商品搭配中。McAuley 等人[6] 与 Veit 等人[16] 尝试将自动化商品搭配生成扩展到更加一般化的场景中，他们通过图片信息为所有类别中的商品生成搭配信息，而不仅仅局限于服饰类别。这两个工作首先利用深度学习技术处理商品的图片信息，然后利用图片中所反映的相似性来衡量商品间的搭配程度。McAuley 等人[6] 通过深度学习将每个商品的图片表示为实数向量，基于这些实数向量通过求解非线性优化（non-linear optimization）问题生成搭配信息。Veit 等人[16] 则利用基于深度学习所构建的端到端（end-to-end）模型直接生成搭配信息。基于深度学习的方法有很多优势：一方面，深度学习模型表现力更强，能够有效地利用大量训练数据，从中学到更为丰富的模式信息；另一方面，深度学习模型无需人工特征工程，尤其是端到端的模型，可以直接以原始数据为输入、以最终目标为输出，几乎不需要任何人工干涉，这极大地增强了模型的实用性，即可以轻松、无缝地加入更多信息乃至切换到全新的任务上。

虽然我们的商品搭配推荐模型也是基于深度学习的方法，但与上述两个工作有两大不同之处：第一，我们利用的是商品标题信息而不是商品图片信息，商品标题具有更少噪声、更多信息等优点且处理起来所需的计算资源更少；第二，我

[2]http://hipsterwars.com

们更关注于商品间的搭配关系，而不是通过图片所反映出的简单相似关系。

2.2 商品销量预测综述

无论是在传统商务中还是在现代电子商务中，商品销量预测一直有着重要的地位。传统销量预测方法主要基于时序分析技术，通过分析历史销量来预测未来销量[17][7]。时序分析技术主要利用历史数据点的线性组合来预测未来的销量，根据组合方式可以将其分为AR（autoregressive）、I（integrated）与MA（moving average）三个大类。将这三大类融合到一起可以形成更为通用的模型ARMA（autoregressive moving average）与ARIMA（autoregressive integrated moving average）等，它们往往可以取得更好的预测效果[18]。当用时序分析技术对销量进行预测时，因为模型只以历史销量数据为输入且用历史销量的线性组合来预测未来销量，所以只适用于那些销量稳定或者销量变化规律明显的商品[8]。

电商的商业环境更为动态而复杂，商品销量变化规律更不明显，时序分析技术在这种场景下预测所能取得的准确性非常有限。从另外一个角度来看，电子商务中大量数据可以被轻松地收集起来并加以利用，如浏览次数（PV）、浏览人数（UV）与价格（PAY）等，将这些数据考虑到模型中能够有效地提升销量预测的准确性。Kulkarni等人[19]利用搜索引擎日志数据来提升销量预测的准确性，他们的基本假设是商品的销量与商品被搜索次数正相关，类似于谷歌对流感爆发与搜索关键字之间关系的研究工作[20]，因此把商品被搜索次数引入预测模型可以提升销量预测的准确性。Ramanathan等人[21]利用供应链等相关信息来提升销量预测的准确性，通过线性回归模型分析了多种影响市场需求的因素并从中选出对销量预测最为有用的若干因素，进一步建立被称为RDM（reference demand model）的预测模型。Yeo等人[22]则做了更加精细化的预测——直接预测某个用户是否会买某个商品。他们通过分析消费者的历史浏览行为数据来预测其将会购买的商品，首先从消费者历史浏览行为数据中提取若干特征，其次利用这些特征构建名为B2P（browsing to purchase）的模型，最后通过这一模型预测某个用户是否会

购买某个商品。如果想利用该模型预测某个商品的销量，需要综合所有用户是否会购买这件商品的预测结果。

上述将更多数据引入模型的方法都是针对特定应用场景而设计的，难以扩展到更广泛的应用场景当中。另外，它们都依赖人工特征工程从数据中提取相关特征，而特征工程往往是个耗时、耗力的工作且要求进行特征工程的人员拥有特定领域的专业知识。这些问题的根本原因是传统数据挖掘方法无法自动重组原始数据，即无法从原始数据中自动寻找起决定作用的因素并进一步将其提取成有效特征。特征学习（feature learning）可以自动化地从原始数据中提取有效特征，消除对人工特征工程的依赖，而深度学习（deep learning）是最常用的特征学习方法之一[23]。近些年来，基于深度学习的方法在诸多领域取得了最好的效果[9][24][25]。深度学习又被称为神经网络（neural networks），其灵感来源于神经系统：将神经元抽象为结点，将突触抽象为边，结点通过边连接在一起。神经网络通过介于输入层与输出层之间的连接关系刻画函数，边的权重便是所刻画函数的参数。

深度学习在介于输入层与输出层之间的隐含层中从原始数据中提取有效特征，然后将这些特征应用于后续输出层的分类或拟合问题中。虽然我们的商品销量预测模型也是基于深度学习的方法，但与现有工作有很大的不同。现有方法主要集中于自动化地从诸如图像、语音、文本等非结构化数据中提取有效特征，而我们关注的是如何对结构化时序数据进行合理变换之后，利用深度学习中的卷积神经网络这一架构，从原始数据中自动化地提取有效特征。我们的模型以与商品相关的原始日志数据为输入，以商品接下来一段时间上的总销量为输出，先从日志数据中提取有效特征，然后将这些特征应用于输出层的销量预测当中。

2.3 卷积神经网络综述

2.3.1 发展简史

卷积神经网络（convolutional neural networks）是深度学习中最为重要的框架之一，它可以很好地利用数据中的“时空局部性”这一先验信息。卷积神经网络最早由 LeCun 等人于 1998 年提出[3]，然后长时间未出现在人们的视线当中。它最近的复兴很大程度上要归功于 AlexNet[9] 在 2012 年 ILSVRC 比赛（ImageNet Large Scale Visual Recognition Challenge）上赢的冠军且误差降低量与传统方法相比非常可观。之后它席卷了计算机视觉（computer vision)[26]、语音识别（speech recognition)[27] 与自然语言处理（natural language processing)[28] 等诸多领域的大部分问题，甚至在人类引以为傲的的智力游戏“围棋”上帮助计算机战胜了世界最强手李世乭[10]。

2.3.2 技术详述

卷积神经网络主要包括卷积层（convolution）、激活函数（activation function）、池化层（pooling）、多重映射（multiple feature maps）与全连接层（full connection）等组件，接下来对这些组件做一些详细的说明。

2.3.2.1 卷积层

卷积层可以看做一种特殊的线性操作，其目标是提取局部模式规律。在本文中，我们使用的是一种被称为一维卷积（one-dimensional convolution）的特殊卷积操作。具体而言，一维卷积是涉及两个向量 $\mathbf{f} \in \mathbb{R}^m$ 和 $\mathbf{s} \in \mathbb{R}^{|s|}$ 的一种操作。其中，向量 $\mathbf{f}$ 是大小为 m 的过滤器（filter），向量 $\mathbf{s}$ 是长度为 $|\mathbf{s}|$ 的序列（sequence）。该操作将过滤器 $\mathbf{f}$ 沿着序列 $\mathbf{s}$ 与它的每个长度为 m 的子序列做点积（dot product）运算，然后得到一个新序列 $\mathbf{c}$：

$$\mathbf{c}_j = \mathbf{f}^{\mathrm{T}} \mathbf{s}_{j-m+1:j}. \qquad \text{公式（2.1）}$$

在实践中，经常要向点积所得到的结果再加上一个偏置（bias）项 b：

$$\mathbf{c}_j = \mathbf{f}^{\mathrm{T}}\mathbf{s}_{j-m+1:j} + b. \qquad \text{公式（2.2）}$$

根据索引 j 所允许的范围不同，卷积操作可分为两种：窄卷积与宽卷积。窄卷积中索引 j 的取值范围为 $[m, |\mathbf{s}|]$，得到的新序列为 $\mathbf{c} \in \mathbb{R}^{|\mathbf{s}|-m+1}$；宽卷积中索引 j 的取值范围为 $[1, |\mathbf{s}| + m - 1]$，得到的新序列为 $\mathbf{c} \in \mathbb{R}^{|\mathbf{s}|+m-1}$，当 $i < 1$ 或 $i > s$ 时 $\mathbf{s}_i$ 的值被填充为 0。相对窄卷积，宽卷积有很多优点[25]：宽卷积对序列中的每个值给予相同的关注，不会轻视接近序列边缘的值；另外，即使在 $|\mathbf{s}| < m$ 的情况下，宽卷积也能生非空的新序列 $\mathbf{c}$。因此，本文中的所有模型均使用宽卷积。

很多情况下，卷积层的输入往往不只是一个由多个数值所组成的序列，而是一个由多个向量所组成的序列，其中每个向量的维度为 d，即形成矩阵 $\mathbf{S} \in \mathbb{R}^{d\times|\mathbf{s}|}$。当把一维卷积应用于矩阵 $\mathbf{S}$ 时，需要的是一个由 d 个长度为 m 的过滤器所组成的过滤器组（filter bank）$\mathbf{F} \in \mathbb{R}^{d\times m}$ 与一个由 d 个偏置所组成的偏置组（bias bank）$\mathbf{B} \in \mathbb{R}^{d}$。$\mathbf{S}$ 的每行是一个由多个数值所组成的序列，$\mathbf{F}$ 的每行是一个过滤器。先将 $\mathbf{S}$ 的每个行序列分别与 $\mathbf{F}$ 的每个行过滤器进行卷积，然后再加上 $\mathbf{B}$ 中对应的偏置便得到一个新矩阵 $\mathbf{C} \in \mathbb{R}^{d\times(|\mathbf{s}|+m-1)}$：

$$\mathrm{conv}(\mathbf{S}, \mathbf{F}, \mathbf{B}) : \mathbb{R}^{d\times|\mathbf{s}|} \to \mathbb{R}^{d\times(|\mathbf{s}|+m-1)}, \qquad \text{公式（2.3）}$$

其中，过滤器组 $\mathbf{F}$ 与偏置组 $\mathbf{B}$ 是模型训练过程中需要优化的参数，过滤器大小 m 是模型的超参数。

2.3.2.2 激活函数

如果神经网络中只有线性变换，则最终只能形成线性函数。为了可以对非线性函数建模，要将非线性激活函数（activation function）$\alpha(\cdot)$ 应用于很多层输出结果的每个元素上，如卷积层，然后得到一个新矩阵 $\mathbf{A} \in \mathbb{R}^{d\times(|\mathbf{s}|+m-1)}$：

$$\alpha(\mathbf{C}) : \mathbb{R}^{d\times(|\mathbf{s}|+m-1)} \to \mathbb{R}^{d\times(|\mathbf{s}|+m-1)}. \qquad \text{公式（2.4）}$$

常见的 $\alpha(\cdot)$ 包括 S 函数（sigmoid）、双曲正切（tanh）与修正线性单元（relu）等。修正线性单元克服了 S 函数及双曲正切的一些缺点，在实际应用中最为常见[29]。另外还考虑到修正单元具有形式简单、计算量少等优点，本文中的所有模型均使用修正线性单元作为激活函数。

结合激活函数与公式（2.2）可以看出，偏置 b 扮演着阈值的角色——控制着哪些单元会被激活及其被激活的程度。

2.3.2.3 池化层

池化层（pooling）将输入的信息进行聚合，旨在使模型对输入数据中的微小变化更加鲁棒。对于向量 $\mathbf{a} \in \mathbb{R}^{|\mathbf{a}|}$，经典的池化操作（参数为 k）会依次将 $\mathbf{a}$ 中每 k 个值聚合成一个值：

$$\text{pooling}(\mathbf{a}) : \mathbb{R}^{|\mathbf{a}|} \to \mathbb{R}^{\lceil |\mathbf{a}|/k \rceil}. \qquad \text{公式（2.5）}$$

当把经典池化操作应用于矩阵 $\mathbf{A}$ 时，$\mathbf{A}$ 的每一行会被分别池化，然后得到一个新矩阵 $\mathbf{P} \in \mathbb{R}^{d \times \lceil |\mathbf{a}|/k \rceil}$：

$$\text{pooling}(\mathbf{A}) : \mathbb{R}^{d \times |\mathbf{a}|} \to \mathbb{R}^{d \times \lceil |\mathbf{a}|/k \rceil}. \qquad \text{公式（2.6）}$$

根据信息聚合方式不同可以将池化操作分为两种：平均化（average）与最大化（max），其中最大化池化操作在实际应用中使用最为广泛。最近，出现了一种名为 K-最大化池化（K-max pooling）的新池化方式[25]，它会从向量 $\mathbf{a}$ 中选取 k 个最大值组成新向量，在新向量中保留它们在输入向量中原有相对位置：

$$\text{k-pooling}(\mathbf{a}) : \mathbb{R}^{|\mathbf{a}|} \to \mathbb{R}^{k}. \qquad \text{公式（2.7）}$$

与经典的最大化池化不同，K-最大化池化可以处理不同长度的输入，得到相同长度的输出。当把 K-最大化池化应用于矩阵 $\mathbf{A}$ 时，分别为 $\mathbf{A}$ 的每一行选取 k 个最大值，然后得到一个新矩阵 $\mathbf{P} \in \mathbb{R}^{d \times k}$：

$$\text{k-pooling}(\mathbf{A}) : \mathbb{R}^{d \times |\mathbf{a}|} \to \mathbb{R}^{d \times k}. \qquad \text{公式（2.8）}$$

无论在哪种池化操作中，都存在一个参数 k，它是模型的超参数。

2.3.2.4 多重映射

对原始输入应用了卷积、非线性激活函数与池化这组操作后，可以得到一阶表达（representation）。通过反复应用上述操作来构建更深的神经网络，可以得到对原始数据的更高阶表达。另外，第 i 阶表达中可以计算 K_i 个不同的表达 $\mathbf{P}_1^i, \cdots, \mathbf{P}_{K_i}^i$。每个表达 $\mathbf{P}_j^i$ 通过两步计算得到：首先，对每个 $i-1$ 阶表达应用卷积操作（过滤器组为 $\mathbf{F}_{j,k}^i$，偏置组为 $\mathbf{B}_{j,k}^i$）后，把得到的多个卷积结果累加在一起；然后，对累加后的结果应用非线性激活函数以及池化操作。经过这两步计算后得到 $\mathbf{P}_j^i$:

$$\mathbf{P}_j^i = \text{pooling}(\alpha(\sum_{k=1}^{K_{i-1}} \text{conv}(\mathbf{P}_k^{i-1}, \mathbf{F}_{j,k}^i, \mathbf{B}_{j,k}^i))). \qquad \text{公式（2.9）}$$

2.3.2.5 全连接层

全连接层（full connection）是一个线性操作，它将最高层的所有表达浓缩到一个向量中。如果最多有 h 阶表达，即最高阶表达为 $\mathbf{P}_1^h, \cdots, \mathbf{P}_{K_h}^h$（假设 $\mathbf{P}_k^h \in \mathbb{R}^{d \times p}$），首先通过拉平（flat）与拼接（concatenate）操作将它们中的所有值合并到一个向量 $\mathbf{p} \in \mathbb{R}^{K_h \times d \times p}$ 中，然后通过矩阵 $\mathbf{H} \in \mathbb{R}^{(K_h \times d \times p) \times n}$ 对该向量进行变换，最后应用非线性激活函数:

$$\hat{\mathbf{x}} = \alpha(\mathbf{p}^{\mathrm{T}}\mathbf{H}), \qquad \text{公式（2.10）}$$

其中 $\hat{\mathbf{x}} \in \mathbb{R}^n$ 可以看做由从原始数据中自动化提取出来的特征所组成的向量，矩阵 $\mathbf{H}$ 是模型训练过程中需要优化的参数，最终向量 $\hat{\mathbf{x}}$ 的维度 n 是模型的超参数。

2.4 本章小结

结合本文的研究重点，本章主要围绕着商品搭配推荐和商品销量这两个研究方向的相关工作进行了归纳与总结。在阐述这些相关研究工作的同时，我们对现有工作的特点和存在的不足进行了具体的讨论，并简单地提到了本文针对相应技术内容提出的创新性思路。另外，本章最后还深入调研和详细介绍了卷积神经网络的技术细节。通过本章对相关研究领域综合性的介绍与分析，我们为本文后续内容的展开奠定了坚实的理论与技术基础。

第 3 章 基于深度样式匹配的商品搭配推荐

3.1 背景介绍

商品搭配信息在很多商业场景中都有应用，比如根据用户已经购买的商品向其推荐可能购买的商品，或直接向用户推荐商品购买套装等，图3.1是衣服套装示例。生成这些信息最直接的做法是人工标定：让若干搭配达人来手工标定哪些商品是相搭配的。在传统商业中商品数量较少，这一方法具有较高的可行性；而电商中商品数量巨大，虽然可以通过众包（crowdsourcing）等方式实现大规模标注，但成本会非常高。如何自动化地生成商品间搭配信息，无论是在学术界还是在工业界，一直都是个备受关注的问题。

图 3.1 衣服套装：左侧是查询商品，右边是与之相搭配的商品

自动化生成商品搭配信息的最经典算法是“频繁项集挖掘”（frequent item-set mining）[4]，它从商品的历史购买记录中挖掘出商品间的搭配关系，其基本假设是经常被共同购买的两个商品往往相搭配，比如最经典的案例：“啤酒”与“尿布”。然而频繁项集挖掘算法有一个致命的缺点，那就是冷启动（cold-start）问题，即无法

为没有历史购买记录的新商品生成搭配信息[5]。而电商中的商品更新周期短，很大一部分都是没有历史购买记录的新商品。

基于内容的商品自动搭配算法可以直接利用商品的描述信息生成搭配，因为电商中所有的商品都有对应的描述信息，所以这些算法从根本上克服了“冷启动”问题。如 McAuley 等人[6] 与 Veit 等人[16] 利用商品的描述图片计算商品间的搭配程度，他们以两个商品图片间视觉上的相似性来衡量两个商品的搭配程度。虽然图片间视觉上的相似性在一定程度上可以用来衡量两个商品的搭配程度（尤其是“服饰”类目），但视觉相似并不完全等同于相搭配；另外，处理图片所需的计算量大，且图片经常包含大量噪声，比如图3.2是淘宝[1] 上的一张典型图片，它本是用于展示裤子，但也包含了大衣、鞋以及非常复杂的背景，这些信息都会对模型造成干扰。

图 3.2 淘宝上一张用于展示裤子的图片

为了克服现有方法的不足，我们设计了一个基于卷积神经网络的全新方法。我们观察到商家为了商品更容易被用户通过搜索引擎访问，会把商品所有的重要属性信息放在标题当中。因此商品标题不但包括商品的外观信息，还包括商品的类目、品牌与适合人群等信息。考虑到商品间的搭配本质上是商品属性间的搭配，利用商品标题组成的短文本对能够更好地为商品间的搭配关系建模。

[1]http://taobao.com

为文本建模的传统方法依赖于人工特征工程[30]，这一过程依赖于很多额外的工具，比如语法解析器、词法解析器与特定领域数据库等。深度学习技术可以自动化地从大量原始数据中抽取有效特征，而卷积神经网络是深度学习中最为重要的框架之一。我们设计了一个对拍卷积神经网络（siamese convolutional neural network)，这个对拍神经网络以两个商品标题所组成的短文本对作为输入：首先，通过卷积神经网络将每个商品的标题表示为一个多维实数向量；然后，将上述实数向量映射到特定的“样式空间”（style space）之中；最后，在样式空间中计算两个商品间的搭配分数。此外，我们还利用最近邻搜索（nearest neighbor search）技术针对算法在实际的推荐应用场景中进行加速。在来自淘宝与亚马逊的两个大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品搭配生成质量。

3.2 算法设计

给定待查询商品集合 $Q = \{q_1, \cdots, q_m\}$ 和候选搭配商品集合 $C = \{c_1, \cdots, c_n\}$，每个查询商品 $q_i \in Q$ 与候选搭配商品集合的搭配关系表示为 $\{y_{i_1}, \cdots, y_{i_n}\}$，即如果 $c_j \in C$ 与 $q_i \in Q$ 相搭配则标定为 $y_{i_j} = 1$，否则标定为 $y_{i_j} = 0$。我们要建立一个模型来计算 q_i 与 c_j 之间的搭配概率：

$$P(y = 1|q_i, c_j) = f(\phi(q_i, \theta_1), \phi(c_j, \theta_1), \theta_2), \qquad \text{公式（3.1）}$$

其中，$\phi(\cdot)$ 是短文本模型，它将短文本映射为实数向量；然后通过 $f(\cdot)$ 在样式空间中计算两个商品间的搭配概率；θ_1 与 θ_2 是在训练过程中需要优化的参数向量。

3.2.1 基于卷积神经网络的短文本模型

我们通过 $\phi(\cdot)$ 为短文本建模，它是一个卷积神经网络，如图3.3所示。

我们的模型以短文本 $\mathbf{s}$ 为输入，它可以看做是个由单词组成的序列：$[s_1, \cdots, s_{|\mathbf{s}|}]$，其中每个单词 s_i 来源于字典 V。首先，将每个单词 s_i 映射为词向量 $\mathbf{w}_i \in \mathbb{R}^d$，这

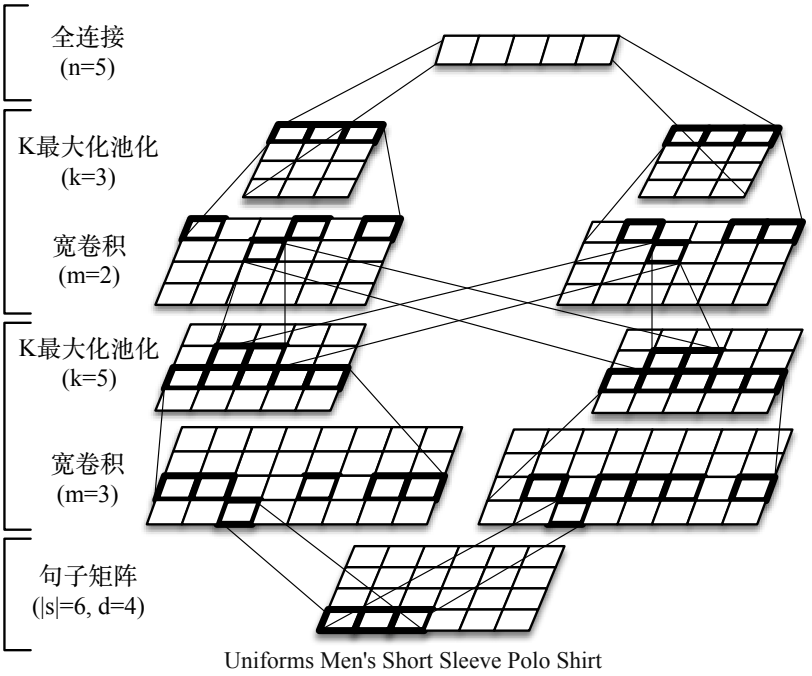


图 3.3 用卷积神经网络将短文本映射为实数向量

些词向量来源于同一个词嵌入矩阵（word-level embedding matrix）$\mathbf{W} \in \mathbb{R}^{d \times |V|}$；然后，为输入短文本 **s** 构建以下句子矩阵（sentence matrix）$\mathbf{S} \in \mathbb{R}^{d \times |\mathbf{s}|}$：

$$\mathbf{S} = \begin{bmatrix} | & | & | \\ \mathbf{w}_1 & \cdots & \mathbf{w}_{|\mathbf{s}|} \\ | & | & | \end{bmatrix}, \qquad \text{公式（3.2）}$$

其中，矩阵中第 i 列是短文本 **s** 第 i 个单词的词向量 $\mathbf{w}_i$。词嵌入矩阵 **W** 是模型训练过程中需要优化的参数，嵌入维度 d 是模型的超参数。

为了实现用向量表示短文本，我们在句子矩阵上应用卷积神经网络，也就是在句子矩阵 **S** 上应用卷积、非线性激活函数、池化与全连接等操作。

以图3.3为例，对于输入的短文本 **s**=[Uniforms, Men's, Short, Sleeve, Polo, Shirt]，经过如下几个步骤表示为实数向量：

1. 将其表示为嵌入维度为 4 的句子矩阵 $\mathbb{R}^{4 \times 6}$；

2. 对句子矩阵应用卷积操作（过滤器组大小为 4×3，偏置组大小为 4）及激活函数后，得到一个大小为 4×8 的新矩阵；

3. 应用 $k = 5$ 的 K 最大化池化操作后，得到一个大小为 4×5 的新矩阵；

4. 在生成一阶表达的过程中，使用两对不同的过滤器组 $\mathbf{P}_1^1 \in \mathbb{R}^{4\times5}$、$\mathbf{P}_2^1 \in \mathbb{R}^{4\times5}$ 及偏置组 $\mathbf{B}_1^1 \in \mathbb{R}^{4\times1}$、$\mathbf{B}_2^1 \in \mathbb{R}^{4\times1}$，生成了两个不同的一阶表达；

5. 在生成二阶表达的过程中，使用另外两对不同的过滤器组 $\mathbf{P}_1^2 \in \mathbb{R}^{4\times5}$、$\mathbf{P}_2^2 \in \mathbb{R}^{4\times5}$ 及偏置组 $\mathbf{B}_1^2 \in \mathbb{R}^{4\times1}$、$\mathbf{B}_2^2 \in \mathbb{R}^{4\times1}$，生成了两个不同的二阶表达；

6. 在生成二阶表达之后，通过大小为 $n = 5$ 的全连接将短文本最终表示为长度为 $n = 5$ 的向量。

3.2.2 通过对拍网络在样式空间中实现搭配

我们通过函数 $f(\cdot)$ 来计算两个商品间的搭配程度，其中 $f(\cdot)$ 是一个对拍卷积神经网络（siamese convolutional neural network），结构如图3.4所示。

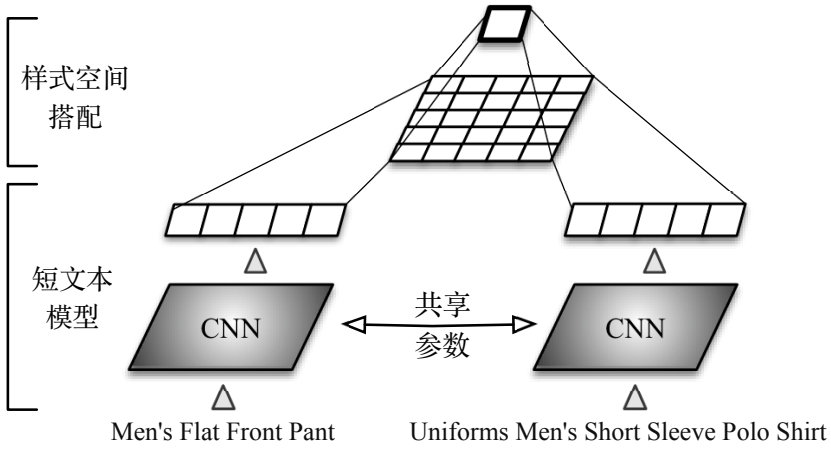


图 3.4 用对拍卷积神经网络构建搭配函数

对拍神经网络的思想最早由 Hadsell 等人于 2006 年引入[31]，然后被广泛地应用于距离度量学习中（distance metrics learning）。如何将两个神经网络组成对拍

结构则需要根据实际应用场景进行设计，当我们设计搭配算法的对拍部分时，着重考虑了它的可扩展性，这在实际应用场景中极为重要。

3.2.2.1 样式空间

对于两个商品 q 与 c，通过卷积神经网络分别将它们的标题映射为实数向量 $\mathbf{x}_q \in \mathbb{R}^n$ 与 $\mathbf{x}_c \in \mathbb{R}^n$ 后，我们通过如下方式计算两个商品间的搭配程度：

$$\begin{aligned} P(y=1|q,c) &= \sigma(\mathbf{x}_q^{\mathrm{T}}\mathbf{M}\mathbf{x}_c + b) \\ &= \frac{1}{1+e^{-(\mathbf{x}_q^{\mathrm{T}}\mathbf{M}\mathbf{x}_c+b)}}, \end{aligned} \qquad \text{公式（3.3）}$$

其中 $\mathbf{M} \in \mathbb{R}^{n\times n}$ 是一个矩阵，b 是一个标量。我们把 $\mathbf{M}$ 称为搭配矩阵（compatibility matrix），把它所张成的空间称为样式空间（style space）。$\mathbf{x}_q$ 经过线性变换 $\mathbf{x}'_q = \mathbf{x}_q^{\mathrm{T}}\mathbf{M}$ 后，$\mathbf{x}'_q$ 代表的是哪种样式与 q 最相搭。然后寻找那些与 $\mathbf{x}'_\mathbf{q}$ 线性核距离（linear kernel distance）最近的商品，这些商品便是与 q 最相搭的。从另外一个角度来看，公式（3.3）中的 $\mathbf{x}_q^{\mathrm{T}}\mathbf{M}\mathbf{x}_c$ 可以看作噪声信道（noisy-channel）模型[32][33]。

矩阵 $\mathbf{M}$ 与 b 是模型训练过程中需要学习的参数向量。

3.2.3 通过 K 近邻搜索在推荐场景中实现加速

在推荐应用场景中，一般有一个待查询商品集合 $Q = \{q_1, q_2, \cdots, q_m\}$ 与一个候选推荐商品集合 $C = \{c_1, c_2, \cdots, c_n\}$，其中待查询商品集合往往较小而候选推荐商品集合往往较大。对于每个待查询商品 q_i，要从候选推荐集合 C 中选出 K 个与之最为搭配的商品并按搭配程度排序。但是，为每对商品 (q_i, c_j) 计算搭配程度后再进行选择和排序这种做法非常低效，甚至在 C 非常大的情况根本不可行。我们的方法可以很容易地扩展到这种需要处理大数据的推荐场景中。

给定商品 q 与 c，我们通过公式（3.3）计算它们之间的搭配程度。注意到函数 $\sigma(\cdot)$ 是单调递增且 b 是一个学到的固定常数，则对商品 q 以及两个候选商品 c_1, c_2

有:

$$\begin{aligned} P(y=1|q,c_1) &\leq P(y=1|q,c_2) \\ \Leftrightarrow \mathbf{x}_q^{\mathrm{T}}\mathbf{M}\mathbf{x}_{c_1} &\leq \mathbf{x}_q^{\mathrm{T}}\mathbf{M}\mathbf{x}_{c_2} \end{aligned}. \qquad \text{公式（3.4）}$$

基于上述性质，我们把从候选推荐集合 C 中查询与商品 q 最搭配的 K 个商品这一原始问题转换为一个新问题：从 $\{\mathbf{x}_{c_1},\cdots,\mathbf{x}_{c_n}\}$ 中查询 $\mathbf{x}'_q$（$\mathbf{x}'_q=\mathbf{x}_q^{\mathrm{T}}\mathbf{M}$）在线性核距离下的 K 近邻，这一问题被称为 MIPS（Maximum Inner Product Search）。有很多方法可以有效地处理大数据下的 MIPS 问题，比如基于树的方法[34] 与基于哈希的方法[35][36] 等。

3.3 模型训练

我们训练模型使其能生成训练集 $\mathcal{R}$ 的可能性最大，即最大化似然概率（maximizing the likelihood）。在训练集 $\mathcal{R}$ 中：$r_{ij}\in\mathcal{R}$:

$$r_{ij}=\begin{cases} 1 & \text{，商品 } i \text{ 与 } j \text{ 相搭;} \\ 0 & \text{，其他。} \end{cases} \qquad \text{公式（3.5）}$$

最大化似然概率等同于最小化二元交叉熵损失（binary-cross entropy loss）:

$$L=-\sum_{r_{ij}\in\mathcal{R}}\left[r_{ij}\log(p)+(1-r_{ij})\log(1-p)\right], \qquad \text{公式（3.6）}$$

其中 $p=P(y=1|i,j)$。

整个网络需要优化的参数为公式（4.1）中的 θ_1、θ_2:

$$\theta_1=\{\mathbf{W},\mathbf{F},\mathbf{B},\mathbf{H}\}、\ \theta_2=\{\mathbf{M},b\}. \qquad \text{公式（3.7）}$$

它们分别是词嵌入矩阵 **W**、过滤器组 **F**、偏置组 **B**、变换矩阵 **H**、搭配矩阵 **M** 及偏置 b。注意，我们需要同时优化多个不同的过滤器组与偏置组。

3.3.1 通过舍弃操作对模型进行正则化

神经网络可以学习非常复杂的函数且非常容易产生过拟合现象，为了防止过拟合我们使用了一种名为舍弃（dropout）的操作[37]。舍弃操作应用于公式（2.10）中的向量 **p**，具体而言是在计算过程中将向量 **p** 中的每个元素以 p 的概率设为 0，从而防止特征间相互适应（co-adaptation），其中舍弃概率 p 是模型的超参数。Goodfellow 等人[23] 认为舍弃操作近似等价于模型平均（model averaging），而模型平均是机器学习中用来提升模型泛化能力最为有效的方法之一。

3.3.2 模型中超参数的设定

在我们的模型中，涉及到的所有超参数设定如下：词嵌入维度为 $d = 100$，一阶表达中过滤器大小为 $m = 3$、K 最大化池化中 $k = 5$，二阶表达中过滤器大小为 $m = 2$、K 最大化池化中 $k = 3$，最终用于表示短文本的向量的维度为 $n = 100$，舍弃操作中舍弃概率为 $p = 0.2$；另外，一阶表达中同时学习 $K_1 = 100$ 种不同表达方式，二阶表达中同时学习 $K_2 = 100$ 种不同表达方式。

3.3.3 通过非监督模型对词嵌入进行初始化

虽然我们的模型可以直接求得词嵌入矩阵 **W**，但利用非监督模型[38] 初始化 **W** 无论是对收敛速率还是对最终结果都有帮助[39]。在初始化过程中，如果某单词未出现在非监督模型的结果中，我们以均匀分布 $U[-0.05, 0.05]$ 对其词向量的每一维进行随机初始化。

3.3.4 求解模型的最优参数

我们利用随机梯度下降算法（stochastic gradient descent）优化模型，通过向后传导（back propagation）的方式更新参数，更新方法采用的是 Adamax 规则[40]。模型训练时每次读取 256 个样本，读完所有样本记为 1 个周期，总共训练 20 个

周期。我们使用 GPU 加速计算，利用 Python 语言及基于 Theano[41] 的 Kears[2]框架实现算法，在单个 NVIDIA K2200 GPU 上每分钟可以处理 42.8 万个文本对。

3.4 实验验证

为了验证我们提出模型的有效性，我们分别在来自淘宝与亚马逊的两个大规模数据集上做了验证实验。

3.4.1 数据集

3.4.1.1 淘宝数据集

该数据集收集于淘宝[3]，由阿里巴巴集团提供[4]。它仅包含“服饰”类目，囊括 6.1 万商品以及它们之间 40.6 万搭配关系，这些搭配关系是由专门的搭配专家人工标定的。在这个数据集中，每个商品有对应的图片和标题，其中标题是分词后的结果。值得注意的是，与英文不同中文中词与词之间没有空格隔开，所以对中文进行自然语言处理之前先要进行分词。

3.4.1.2 亚马逊数据集

该数据集收集于亚马逊[5]，由 McAuley 等人[6] 提供。尽管它包含了多个类目，为了对照我们的方法在两个数据集上的效果，我们更关注于其中的“服饰”类目。在这个类目中，该数据集囊括 66.2 万商品以及它们之间 1.2 千万搭配关系。每个商品有对应的图片、标题和其他一些信息。与淘宝数据集不同，该数据集里的搭配关系不是由人工标定的，而是通过共同购买数据生成的[42]。

[2]http://keras.io

[3]http://taobao.com，淘宝是中国最大的网络零售平台

[4]http://tianchi.aliyun.com/datalab/index.htm

[5]http://amazon.com，亚马逊是世界上最大的网上商店

3.4.2 实验设定

我们的目标是区分搭配关系与非搭配关系，所以将上述数据集中的搭配关系作为正样本（positive），再随机生成若干负样本（negative），正样本与负样本之间的比例为 50：50。生成负样本后，将整个数据集随机划分为训练集、验证集与测试集三部分，三部分的比例为 80：10：10。为了防止对测试集过拟合，我们在验证集上调优模型的超参数。

我们将我们的方法与以下几种方法进行比较，主要分为两大类：视觉方法与文本方法。

3.4.2.1 视觉方法

我们以 Veit 等人的方法[16] 作为用来对比的视觉方法，因为他们的方法也遵从端到端的形式。更具体而言，我们使用的是以 GoogLeNet[26] 为基础的方法，因为在 Veit 等人的论文[16] 中，以 GoogLeNet[26] 为基础的方法在任何场景下都比以 AlexNet[9] 为基础的方法表现好。在淘宝数据集上我们重复了他们的实验，在亚马逊数据集上我们直接使用了 Veit 等人论文[16] 中的结果。

3.4.2.2 文本方法

我们与如下三种文本方法进行对比：

- **NBBW**。朴素贝叶斯（naive bayes）分类器是结合贝叶斯理论与特征间相互独立的假设所得到的分类器。把由两个商品标题所构成的文本对表示为词袋模型（bag of words）之后，在词袋模型上应用朴素贝叶斯分类器便形成对比方法 NBBW（Naive Bayes on Bag of Words）。

- **RFBW**。随机森林（random forest）分类器是由多棵随机化决策树组合而成的集成分类器，它可以为非常复杂的分类面建模。把来自两个商品标题所

构成的文本对表示为词袋模型之后，在词袋模型上应用随机森林分类器便形成对比方法 RFBW（Random Forest on Bag of Words）。

- **RFTM**。主题模型（topic model）可以从文本中挖掘出抽象的“主题”信息。给定商品对 $\{q, c\}$，首先通过 LDA 模型[43] 生成对应的主题表达向量 $\mathbf{x}_q, \mathbf{x}_c$，然后把两个向量拼接成一个向量 $\mathbf{x}_{q,c}$，最后在该向量上应用随机森林分类器便形成对比方法 RFTM（Random Forest on Topic Model）。为了与我们的方法保持一致，将用于表示文本的主题表达向量的维度设为 100，即 $|\mathbf{x}_q| = |\mathbf{x}_c| = 100$，$|\mathbf{x}_{q,c}| = 200$。

在我们的实验中，朴素贝叶斯分类器与随机森林分类器来源于 scikit-learn[44] 工具包。

3.4.3 实验结果

在我们的实验中，图片与文本均未采用任何预处理，下面的结果都是在测试集上测试所得到的结果。

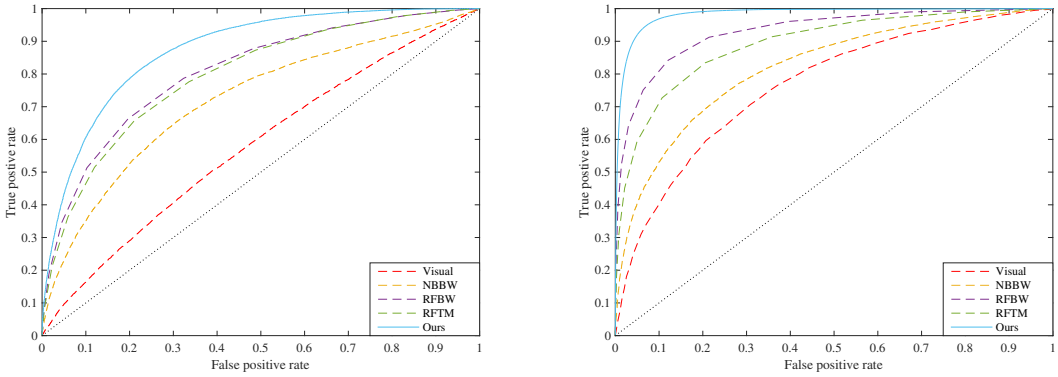
3.4.3.1 对比结果

图3.5是不同方法的 ROC 曲线，表3.1是相应曲线下的面积，即 AUC（Areas Under the Curves）分数。随机分类器的 ROC 曲线是图中的黑色虚线，相应的 AUC 分数为 0.500。可以看到，我们的方法比所有的对比方法表现都要好。

基于视觉的方法在淘宝数据集上表现非常差，那是因为与亚马逊不同，淘宝是 C2C（Consumer to Consumer）平台——对用户所上传的商品图片要求较宽松，很多商品图片都与图3.2类似，其中包含了大量噪声。与商品图片不同，商品标题浓缩了大量信息且所含噪声很少。当使用商品标题时，即使 NBBW 这样的简单方法也可以取得较好的效果；当使用了随机森林这种更复杂的分类器时，取得的效果已经非常有竞争力。使用主题模型后效果反而会变差一些，可见抽象出来

表 3.1 对比方法及我们提出方法的 AUC 分数

方法	淘宝数据集	亚马逊数据集
Visual	0.579	0.770
NBBW	0.712	0.820
RFBW	0.807	0.931
RFTM	0.796	0.893
Ours	**0.891**	**0.983**



(a) 淘宝数据集 (b) 亚马逊数据集

图 3.5 对比方法与我们提出方法的 ROC 曲线

的“主题”并不能很好地为“样式”建模，且在抽取主题的过程中丢失了一些原始信息。上述三种方法都是基于词袋模型的方法，我们方法比它们效果更好是因为我们的方法可以识别句子中的词组甚至更长范围上的模式，这是词袋模型无法办到的。比如，一个商品的标题是“white shirt with blue stripes”，另外一个商品的标题是“blue shirt with white stripes”，虽然它们的词袋模型相同，但与两者相搭配的商品大为不同。

我们方法在亚马逊数据集上的效果要好于其在淘宝数据集上的效果，一方面

是由于亚马逊数据集更大，提供的信息更为全面丰富；另一方面是淘宝数据集中的商品标题事先进行了分词处理，而分词算法的性能会限制最终搭配算法的表现效果。

3.4.3.2 模型调优

在我们的模型中，有很多关键参数：1）词嵌入维度 d；2）文本表达向量的维度 n；3）对词嵌入矩阵 **W** 是否进行初始化。图3.6展示的是在淘宝数据集上，我们的方法在不同设定下，前 10 个训练周期的 AUC 分数。其中标准设定（standard）指的是 $d = 100, n = 100$，且利用非监督方法初始化词嵌入矩阵。

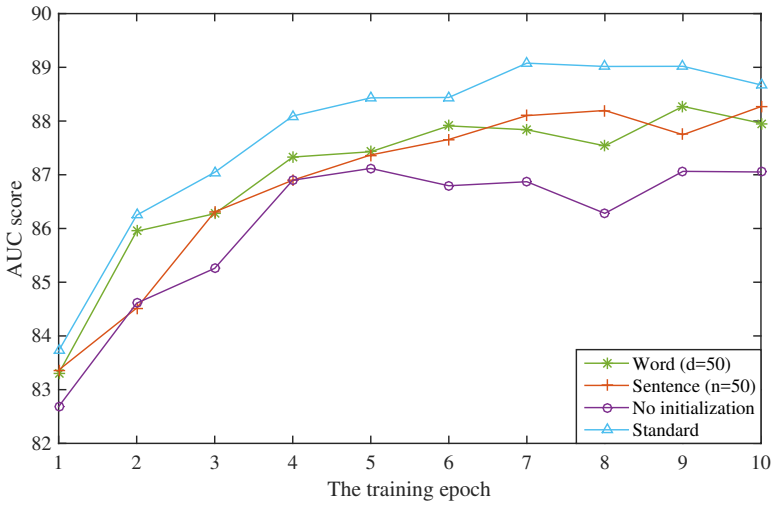
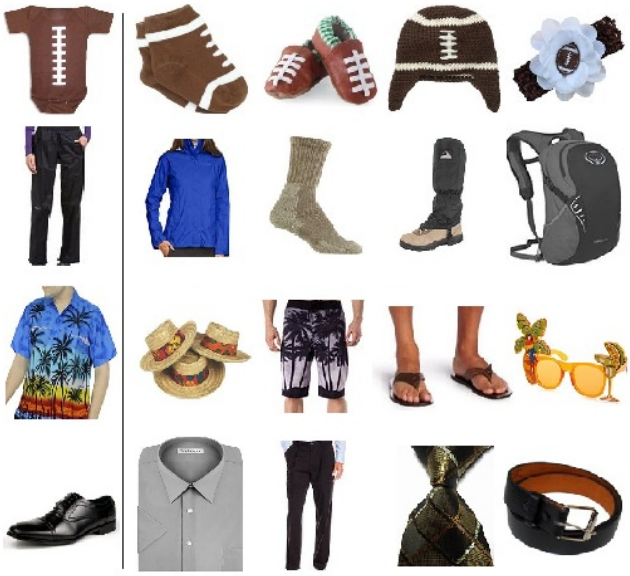


图 3.6 在淘宝数据集上，我们的方法在不同设定下的收敛过程

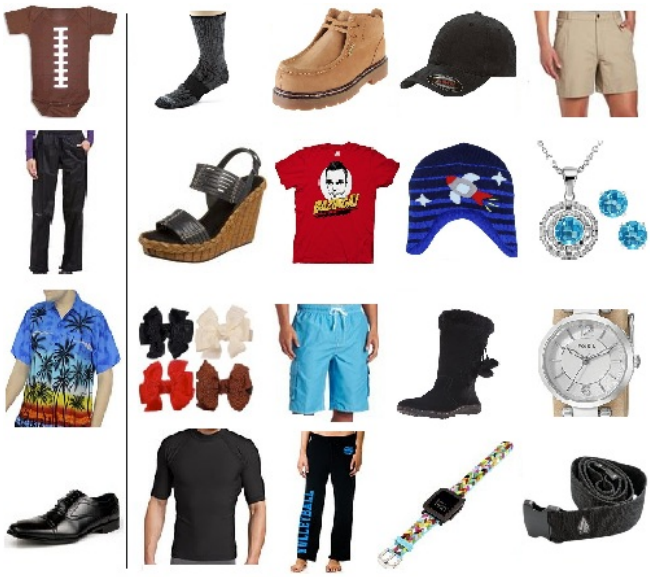
可以看到，更大的 d 和 n 可以带来更好的效果。但更大的 d 和 n 意味着更大的存储及计算开销，也就是说在实际应用中，我们需要在效果与存储及计算开销之间取得一个良好的折中。另外，从实验结果中可以看到，利用非监督方法对词嵌入矩阵 **W** 初始化对收敛速度与最终的搭配效果都有帮助。

3.4.3.3 案例展示

图3.7是通过我们的模型计算出来的若干案例，包括搭配案例与非搭配案例。



(a) 搭配案例: 左侧是查询商品，右侧是最搭配商品



(b) 非搭配案例: 左侧是查询商品，右侧是最不搭配商品

图 3.7 通过我们的模型计算出来的若干搭配案例与非搭配案例

可以看到，我们的模型不但考虑了视觉因素，还考虑了诸如类目、适合季节与人群等因素。以分别在子图（a）与（b）中的第 3 行、第 4 列为例，一件男式夏季 T 恤衫与一双男式夏季人字拖鞋最搭配，与一双女式冬季靴子最不搭配。

3.4.4 讨论

3.4.4.1 通用搭配

3.4.3中的结果主要集中在“服饰”类目上，实际上我们的方法可以在更为通用的场景中实现搭配。表3.2是我们的方法在亚马逊数据集中 20 个类目上的搭配效果，可以看到我们的方法在大部分类目上表现优异。另外，我们还尝试了训练一个单模型实现在所有类目上进行搭配，但这个单模型的效果并不令人满意：AUC 分数只有 0.694。

表 3.2 我们的方法在亚马逊数据集中 20 个类目上的 AUC 分数

类目	AUC 分数	类目	AUC 分数
Automotive	0.922	Home & Kitchen	0.949
Baby	0.917	Movies & TV	0.878
Beauty	0.935	Musical Instruments	0.983
Books	0.897	Office Products	0.974
CDs & Vinyl	0.815	Patio Lawn & Garden	0.966
Cell Phones & Accessories	0.969	Pet Supplies	0.972
Digital Music	0.818	Sports & Outdoors	0.912
Electronics	0.948	Tools & Home Improvement	0.952
Grocery & Gourmet Food	0.959	Toys & Games	0.985
Health & Personal Care	0.929	Video Games	0.890

类目之间的差距也反映出了很多有意思的现象：我们的方法在“CDs & Vinyl”与“Digital Music”两个类目上表现较差，那是因为音乐内容非常丰富，远远超出了音乐标题所能包含的信息。与之相反，对于“Musical Instruments”类目，标题包含了足够多的信息，所以我们的方法在该类目上表现良好。总而言之，适合我们方法的类目应满足如下要求：商品标题可以很好地描述商品。

3.4.4.2 共同购买 **vs.** 搭配

淘宝数据集中的搭配关系是由搭配达人手工标注的，同时该数据集还提供了 1400 万条相关商品的历史购买记录。通过对搭配关系的建模与对共同购买记录的挖掘，我们可以探索共同购买与搭配之间的关系。图3.8展示了淘宝数据集中共同购买最为频繁的 5 对商品，每对商品的下方是通过我们模型所计算出来的搭配分数。

图 3.8 淘宝数据集中共同购买最为频繁的 5 对商品及其搭配分数

我们可以看到共同购买所包含的范围更为广泛：不但包含相搭配的商品（如图3.8中的第 2、5 列），还包含可替换的商品（如图3.8中的第 3、4 列）。而第一列中的商品被频繁共同购买可能是由于它们都是热销商品。

3.5 本章小结

在本章中，我们提出了一种新颖的方法对商品间的搭配程度进行建模。该方法的基本假设是电商中大部分商品的重要属性信息均放在商品的标题之中，而商品间的搭配本质上是商品属性间的搭配，所以可以通过对两个商品标题所组成的短文本对进行建模来实现对商品间的搭配搭配程度建模。我们设计了一个对拍卷积神经网络对短文本对进行建模，这一网络先把两个商品标题分别映射为实数向量，然后将实数向量映射到特定的搭配空间中，最终在搭配空间中计算两个商品间的搭配程度。我们的方法以未经预处理的原始标题文本作为输入，以最终的搭配分数作为输出，无需任何人工特征工程。另外，我们的方法还可以通过 MIPS 等最近邻搜索技术扩展到实际的大数据推荐场景中。

接下来，我们还可以在以下几个方面做更深入的研究。首先，我们希望在保持模型简洁的前提下探索更为复杂的文本模型；其次，同时利用商品的图片和标题信息将会非常有趣，比如将图片与标题映射到相同空间之后，实现图片与标题间的互相检索甚至互相搭配等。

第4章 基于深度时序分析的商品销量预测

4.1 背景介绍

电商中动态而复杂的商业环境为商业决策带来了巨大的挑战，而数据挖掘技术可以帮助人们更好地应对这些挑战。商品销量预测是其中重要的技术之一，它可以帮助人们管理人力、物力与仓储等。

销量预测的价值取决于它的准确性，过高估计销量会导致商品积压，过低估计销量会导致商品脱销。在传统商务中所涉及到的商品数量少，销量预测可以直接由人来完成。较大的商家会利用信息系统对历史销量数据进行保存，然后利用历史销量数据辅助人做出更为准确的预测。如果涉及到的商品数量较多，可以利用时序数据分析（time series analysis）技术自动化地对每个商品的销量做出预测[7]。人工销量预测只适用于商品数量较少的场景，时序数据分析技术只适用于销量稳定或变化规律明显的商品[17]。然而，电商中商品数量众多，且商品销量的变化规律更不明显（图4.1所展示的是菜鸟网络[1]中的某个商品的销量变化情况），上述方法都无法有效地应对[8]。

虽然电商中商品销量预测所面临的挑战更大，但与传统商业不同的是大量数据可以被轻松地收集起来并加以利用。除了历史销量数据，还可以在很长一段时间上收集很多其他日志数据，如浏览次数（PV）、搜索次数（SPV）、浏览人数（UV）、搜索人数（SUV）、成交总额（GMV）与价格（PAY）等。利用这些数据可以有效地提升销量预测的准确性，通常的做法是通过监督学习方法建立回归模型（regression models），将这些信息整合到销量预测模型中。

传统监督学习方法依赖于人工特征工程——利用领域知识从可用数据中提取有效的特征[1]。人工提取特征的质与量将极大地影响预测模型的准确性，而提

[1] http://cainiao.com

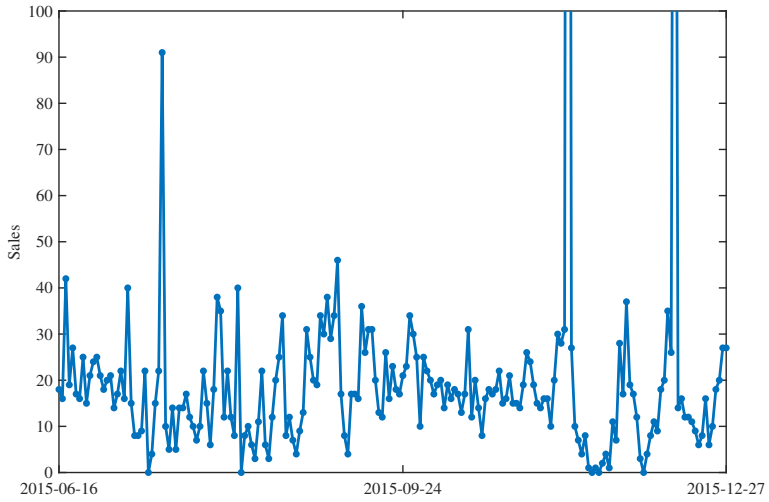


图 4.1 菜鸟网络中某商品的销量变化情况

取有效的特征是个费时、费力的任务。另外，特征工程需要针对特定的应用展开，也就是说当处理新数据或新任务时，要重新进行特征工程。比如，如果有更多的数据可用于商品销量预测，则需要专业人员对这些新数据做完特征工程之后，预测模型才能够有效地利用这些新数据。可见，特征工程限制了模型的可扩展性，使之无法快速、有效地利用电商中与日俱增的可用数据。

特征学习（feature learning）可以取代人工特征工程[2]，它可以从原始输入数据中自动化地提取有效特征，这些特征可进一步用于特定的任务当中。深度学习是最常用的特征学习方法之一，近些年来基于深度学习的方法在诸多领域取得了最好的效果[9][24][25]。深度学习又被称为神经网络，其灵感来源于神经系统：将神经元抽象为结点，将突触抽象为边，结点通过边连接在一起。神经网络通过介于输入层和输出层之间的连接关系刻画函数，边上的权重便是所刻画函数的参数。深度学习在介于输入层与输出层之间的隐含层中自动化地从原始数据中提取特征，然后将这些特征应用于后续输出层的分类或拟合问题中。卷积神经网络（convolutional neural networks）是深度学习中最为重要的框架之一，它可以很好地利用数据中的“时空局部性”这一先验信息[3]。

现有方法主要集中于自动化地从诸如图像[9]、语音[24]、文本[25]等非结构化数据中提取有效特征。在本文中，我们提出了一种新颖的方法，利用卷积神经网络从结构化时序数据（structured time series data）中自动化地提取有效特征。首先，我们结合商品的固有属性信息把与商品相关的原始日志数据转换为特定的“数据框”（data frame）格式，其中原始日志数据包括商品在过去很长一段时间上的销量、价格、浏览次数、浏览人数、搜索次数、搜索人数、收藏人数、加购物车人数等诸多指标；然后，在该数据框上应用卷积神经网络提取有效的特征；最后，在卷积神经网络的最后一层利用这些特征预测商品的销量。此外，我们还利用样本权重衰减（sample weight decay）与迁移学习（transfer learning）等技术进一步提升商品销量预测的准确性。我们的方法以原始日志数据为输入，以最终的销量预测结果为输出，几乎不需要任何人工干涉。在来自菜鸟网络的大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品销量预测的准确性。

4.2 算法设计

对于给定的商品 i 于特定区域 r，我们希望利用一段时间 $[1,T]$ 上与之相关的日志数据 $\mathbf{x}_{ir_t}$，来预测该商品在地区 r 上接下来一段时间 $[T+1,T+l]$ 内的总体销量 y_{ir}。在本文中，用 $\mathbf{x}_{ir_t}$ 表示商品 i 于特定区域 r 内在特定时间点 t 上的商品向量（item vector）。该向量包含商品 i 的 d 维相关信息，如销量（SALE）、浏览次数（PV）、搜索次数（SPV）、浏览人数（UV）、搜索人数（SUV）、成交总额（GMV）与价格（PAY）等。然后，将多个时间点上的 $\mathbf{x}_{ir_t}$ 组合在一起形成商品矩阵（item matrix）$\mathbf{X}_{ir}=[\mathbf{x}_{ir_1},\cdots,\mathbf{x}_{ir_T}]$。另外，用向量 $\mathbf{a}_i$ 表示商品 i 的固有属性集合，包含类别、品牌与供应商等。

我们的目标是构建一个映射函数 $f(\cdot)$，以 $\mathbf{X}_{ir}$ 和 $\mathbf{a}_i$ 为输入来预测 y_{ir}：

$$y_{ir}=f(\mathbf{X}_{ir},\mathbf{a}_i,\theta), \qquad \text{公式（4.1）}$$

其中 θ 是在训练过程中需要优化的参数向量。

4.2.1 将日志时序数据转化为数据框

对每个商品 i，我们需要通过如图4.2所示的方式，根据其固有属性信息将与之相关的日志数据构造成数据框（data frame）。

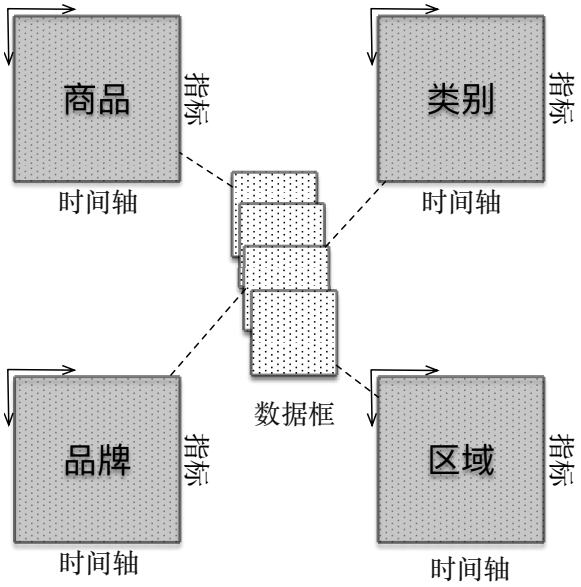


图 4.2 构造数据框

首先，对每个品牌 b、类别 c 与供应商 s，分别计算其于区域 r 内在时间 t 上的品牌向量（brand vector）$\mathbf{x}_{br_t}$、类别向量（category vector）$\mathbf{x}_{cr_t}$ 与供应商向量（supplier vector）$\mathbf{x}_{sr_t}$：

$$\mathbf{x}_{br_t} = \sum_{brand(i)=b} \mathbf{x}_{ir_t}, \qquad \text{公式（4.2）}$$

$$\mathbf{x}_{cr_t} = \sum_{category(i)=c} \mathbf{x}_{ir_t}, \qquad \text{公式（4.3）}$$

$$\mathbf{x}_{sr_t} = \sum_{supplier(i)=s} \mathbf{x}_{ir_t}. \qquad \text{公式（4.4）}$$

然后将上述向量分别组合成品牌矩阵（brand matrix）$\mathbf{X}_{br} = [\mathbf{x}_{br_1}, \cdots, \mathbf{x}_{br_T}]$、类别矩阵（category matrix）$\mathbf{X}_{cr} = [\mathbf{x}_{cr_1}, \cdots, \mathbf{x}_{cr_T}]$ 与供应商矩阵（supplier matrix）$\mathbf{X}_{sr} = [\mathbf{x}_{sr_1}, \cdots, \mathbf{x}_{sr_T}]$。

其次，对每个区域 r, 我们计算在 t 时间上的区域向量（region vector）$\mathbf{x}_{r_t}$：

$$\mathbf{x}_{r_t} = \sum_i \mathbf{x}_{ir_t}. \qquad \text{公式（4.5）}$$

然后将区域向量组合成区域矩阵（region matrix）$\mathbf{X}_r = [\mathbf{x}_{r_1}, \cdots, \mathbf{x}_{r_T}]$。

最后, 对每个商品 i 于区域 r，我们构造数据框 $\mathbf{DF}_{ir}$：

$$\mathbf{DF}_{ir} = [\mathbf{X}_{ir}, \mathbf{X}_{brand(i)r}, \mathbf{X}_{category(i)r}, \mathbf{X}_r]. \qquad \text{公式（4.6）}$$

4.2.2 通过卷积神经网络预测商品销量

我们通过函数 $f(\cdot)$ 对商品的销量进行预测，它是一个卷积神经网络，结构如图4.3所示。

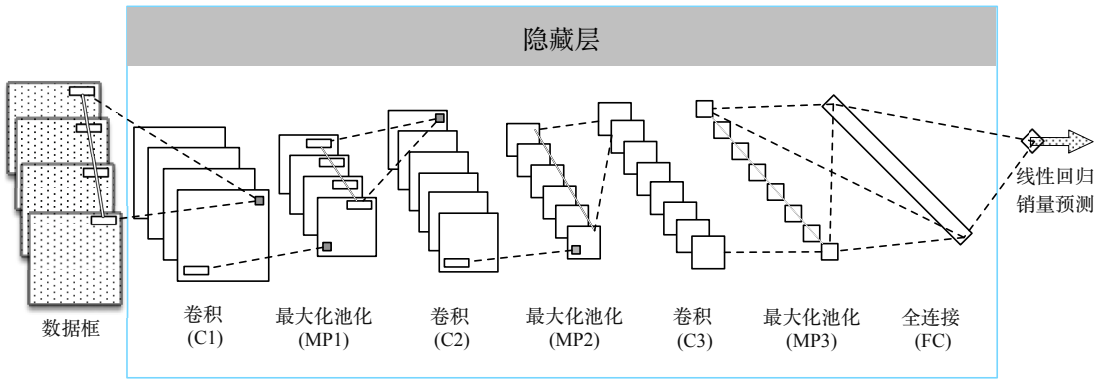


图 4.3 用卷积神经网络进行销量预测

当预测商品 i 于区域 r 的销量时，先要对数据框 $\mathbf{DF}_{ir}$ 应用卷积、非线性激活函数、最大化池化等操作。将上述操作反复应用三次后，便得到对原始日志数据的最高阶表达，然后利用全连接操作把所有的最高阶表达聚合成最终的特征表示向量 $\hat{\mathbf{x}}_{ir}$。得到特征表示向量后，将其作为线性回归的输入来预测商品 i 于区域 r 上的销量 y_{ir}：

$$y_{ir} = [1, \hat{\mathbf{x}}^{\mathrm{T}}] \cdot \mathbf{w}, \qquad \text{公式（4.7）}$$

其中向量 $\mathbf{w}$ 是模型训练过程中需要优化的参数。

4.3 模型训练

我们分别为每个区域建立一个模型——对于每个区域 r，训练一个模型使其在训练集 $\mathcal{D}_r$ 上的均方差 MSE（Mean Squared Error）达到最小：

$$L_r = \frac{1}{|\mathcal{D}_r|} \sum_{ir \in \mathcal{D}_r} (y_{ir} - \hat{y}_{ir})^2, \qquad \text{公式（4.8）}$$

其中 y_{ir} 是商品 i 于区域 r 在时间 $[T+1, T+l]$ 上的真实总销量，$\hat{y}_{ir} = f(\mathbf{X}_{ir}, \mathbf{a}_i, \theta)$ 是对应的预测销量。

整个模型中需要优化的参数为公式（4.1）中的 θ：

$$\theta = \{\mathbf{F}, \mathbf{B}, \mathbf{H}, \mathbf{w}\}. \qquad \text{公式（4.9）}$$

它们分别是过滤器组 **F**、偏置组 **B**、变换矩阵 **H** 及线性回归参数 **w**。注意，我们需要同时优化多个不同的过滤器组与偏置组。

4.3.1 训练样本权重随时间衰减

我们可以通过滑动数据的时间窗口构造很多训练样本，但不同的训练样本重要程度不同：离预测区间越近的点应该有更大的权重。设 sp 为预测区间的起始点，l 为预测区间的长度，ep_{ir} 为数据框 $\mathbf{DF}_{ir}$ 的终止点，显然有 $ep_{ir} \leq sp - l$ 成立。对每个区域 r，我们为训练集 $\mathcal{D}_r$ 中的每个样本分配如下权重：

$$\text{weight}_{ir} = e^{\beta \times (ep_{ir} - sp + l)}, \qquad \text{公式（4.10）}$$

其中 β 是模型的超参数。

然后对每个区域 r，我们最小化模型在训练集 $\mathcal{D}_r$ 上的加权均方差（weighted mean squared error）：

$$L_r^w = \frac{1}{|\mathcal{D}_r|} \sum_{ir \in \mathcal{D}_r} \text{weight}_{ir} \times (y_{ir} - \hat{y}_{ir})^2, \qquad \text{公式（4.11）}$$

其中权重 weight_{ir} 根据公式（4.10）计算得到。

4.3.2 通过迁移学习在区域间共享变化模式

迁移学习旨在将一个问题中模型所学到的知识迁移到另外一个问题中[45]。在本文中，我们希望将一个区域内模型所学到的变化模式迁移到另外一个区域内。虽然商品销量在不同区域内的变化模式有所不同，但与销量相关的某些指标的变化模式在不同区域上有共通之处。比如，虽然中国北方地区要比南方地区更早地售卖棉衣，但在棉衣销量提升之前搜索棉衣的次数会明显增多。

基于此，我们希望预测模型可以先利用所有数据学习通用模式特征，然后再利用特定区域上的数据学习专用模式特征。

首先，我们在整个数据集 $\mathcal{D}$ 上训练我们的神经网络模型，其中：

$$\mathcal{D} = \bigcap_{r} \mathcal{D}_r. \qquad \text{公式（4.12）}$$

然后对每个区域 r，分别将训练集替换为 $\mathcal{D}_r$ 后继续训练调优（fine-tuning)，得到适用于区域 r 的特定模型。

4.3.3 通过舍弃操作对模型进行正则化

神经网络可以学习非常复杂的函数且非常容易产生过拟合现象，为了防止过拟合我们使用了一种名为舍弃（dropout）的操作[37]。舍弃操作应用于公式（2.10）中的向量 **p**，具体而言是在计算过程中将向量 **p** 中的每个元素以 p 的概率设为0，从而防止特征间相互适应（co-adaptation），其中舍弃概率 p 是模型的超参数。Goodfellow 等人[23] 认为舍弃操作近似等价于模型平均（model averaging），而模型平均是机器学习中用来提升模型泛化能力最为有效的方法之一。

4.3.4 模型中超参数的设定

在我们的模型中，涉及到的所有超参数设定如下：一阶表达中过滤器大小为 $m = 7$、最大化池化长度为 $k = 7$，二阶表达中过滤器大小为 $m = 4$、最大池化

长度为 $k=4$，三阶表达中过滤器大小为 $m=3$、最大池化长度为 $k=3$。这样设定超参数是希望一阶表达捕获到星期尺度的模式，二阶表达捕获到月份尺度的模式，三阶表达捕获到季度尺度的模式。

另外，最终的特征表示向量维度为 $n=1024$，权重衰减中的系数为 $\beta=0.02$，舍弃操作中的舍弃概率为 $p=0.2$；每阶表达分别同时计算 128 种不同的表达方式，即 $K_1=K_2=K_3=128$。

4.3.5 求解模型的最优参数

我们利用随机梯度下降算法（stochastic gradient descent）优化模型，通过向后传导（back propagation）的方式更新参数，更新方法采用的是 Adamax 规则[40]。模型训练时每次读取 128 个样本，读完所有样本记为 1 个周期，先在总数据集 $\mathcal{D}$ 上训练 10 个周期，然后分不同模型分别在数据集 $\mathcal{D}_r$ 上再训练 10 个周期，从而为每个区域 r 得到对应的模型。我们对输入的数据框进行了 Z-分数标准化（z-score normalization），这样做无论是对收敛速度还是对最终的预测准确性都有帮助[46]。

我们使用 GPU 加速计算，利用 Python 语言及基于 Theano[41] 的 Kears[2]框架实现算法，在单个 NVIDIA K2200 GPU 上每分钟可以处理 7.3 万样本。

4.4 实验验证

为了验证我们提出模型的有效性，我们在来自菜鸟网络的大规模数据集上做了验证实验。

[2]http://keras.io

4.4.1 数据集

该数据集收集于菜鸟网络[3]，由阿里巴巴集团提供[4]。它包括1814892条记录，囊括5个区域上1963个商品的属性信息及相关日志数据，时间跨度从2014-10-10到2015-12-27。每条日志数据记录了 $d = 25$ 维指标，包括销量（SALE）、浏览次数（PV）、搜索次数（SPV）、浏览人数（UV）、搜索人数（SUV）、成交总额（GMV）与价格（PAY）等。

4.4.2 实验设定

实验中使用的数据框长度为 $T = 84$，预测时段长度为 $l = 7$。对于每个区域 r，我们的目标是利用跨度为 [2015-10-28, 2015-12-20] 的数据框，预测每个商品 i 于该区域在时间段 [2015-12-21, 2015-12-27] 上的总销量，即这些数据样本组成测试集。训练模型时使用的样本数据框的终止点跨度为 [2015-01-01, 2015-12-13]，通过滑动时间窗口得到更多训练数据样本后，将这些数据样本随机地划分为训练集与验证集两部分，两部分的比例为80:20。为了防止对测试集过拟合，我们在验证集上调优模型的超参数。

我们对我们的方法做了如下比较实验，主要分为两大类：对比方法与我们方法的不同设定。

4.4.2.1 对比方法

- **ARIMA**。ARIMA是经典的时序分析方法，在商品销量预测中，它以商品的历史销量数据为输入来预测商品接下来的销量。

- **FE+GBRT**。GBRT（Gradient Boosting Regression Tree）是最常用的回归模型之一，它可以对非常复杂的函数进行建模[47]。当使用它进行销量预测时，

[3]http://cainiao.com，菜鸟网络是中国最大的社会化物流协同平台

[4]http://tianchi.shuju.aliyun.com/competition/information.htm?raceId=231530

我们首先为每个商品人工提取523维特征，包括该商品过去一天的UV、过去3天的平均UV、过去一周的平均UV、过去一个月的平均UV与最近是否降价等信息；然后把这些特征作为输入用GBRT模型来预测该商品接下来的销量。

- **DNN**。DNN是最简单的神经网络结构，它由多层全连接组成。首先，将数据框拉平为向量；然后，通过四层全连接得到最终的特征表示向量，其中每层全连接的维度设为1024；最后，在表示向量上应用线性回归实现销量预测。另外，对最终的表示向量应用了舍弃操作，其中舍弃概率设为$p = 0.2$。

在我们的实验中，ARIMA算法来源于pandas[48]工具包，GBRT算法来源于xgboost[49]工具包。

4.4.2.2 不同设定

- **CNN**。在数据框上应用卷积神经网络是我们方法的基础版本。
- **CNN+WD**。根据公式（4.10）为训练样本分配权重后，训练模型使之最小化加权均方差，这可以提升销量预测的准确性。
- **CNN+WD+TL**。在包含所有训练样本的数据集$\mathcal{D}$上学习通用模型之后，在每个区域r对应的训练数据集$\mathcal{D}_r$上继续训练得到特定模型，这可以进一步提升销量预测的准确性。
- **Single-CNN**。在包含所有训练样本的数据集$\mathcal{D}$上学习通用模型，对每个区域r直接用通用模型预测商品在该区域内的销量。这里并不使用权重衰减技术与迁移学习技术。

在我们的实验中，所有的结果都是在测试集上测试所得到的结果，使用的指标是均方差（MSE）。

4.4.3 实验结果

表4.1是5个区域内测试数据集上实验结果的总体MSE概要，图4.4是5个区域内测试数据集上实验结果的详细箱型图。

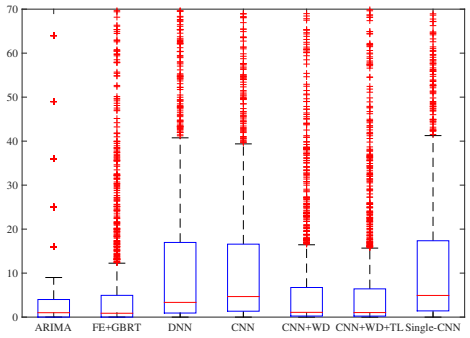
表4.1 所有方法在5个区域内测试数据集上的MSE分数

方法	1	2	3	4	5	平均
ARIMA	104.37	96.68	190.50	397.08	87.18	175.16
FE+GBRT	97.36	83.90	187.06	329.81	82.21	156.07
DNN	97.50	73.55	181.67	347.50	82.17	156.48
CNN	96.98	72.22	151.96	326.39	80.91	145.69
CNN+WD	89.01	56.14	142.27	301.79	75.09	131.86
CNN+WD+TL	**84.30**	**53.40**	**134.92**	**287.31**	**71.19**	**126.22**
Single-CNN	101.92	75.70	159.48	343.54	85.04	153.22

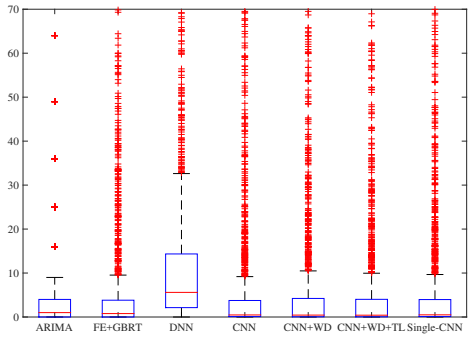
与时序数据分析技术ARIMA相比，经典的数据挖掘方法FE+GBRT可以考虑到更多信息，预测效果更好。虽然DNN是最简单的神经网络架构，但它可以自动化地提取特征。在某些情况下，DNN所提取的特征要比人工提取的特征更为有效，所以在编号为2、3和5的区域上DNN预测效果比FE+GBRT要好。

卷积神经网络可以更好地利用数据中时间维度上的局部性这一先验信息，从而更有效地提取特征，预测效果也随之大幅度提升。最后，样本权重衰减技术与迁移学习技术带来的提升非常可观，综合所有技术后的预测效果非常具有竞争力。另外，从图4.4可以看出，我们的预测模型具有更强的鲁棒性。

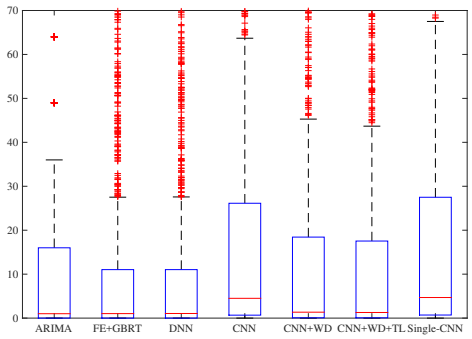
我们还尝试了利用包含所有训练样本的数据集 $\mathcal{D}$ 训练一个单模型，对每个区域 r 直接用这一单模型预测商品在该区域内的销量，这里未使用权重衰减技术与迁移学习技术。从表4.1可以看出，虽然单模型的预测效果较为可观，但与分不同区域训练多个预测模型的效果差距也较为明显。



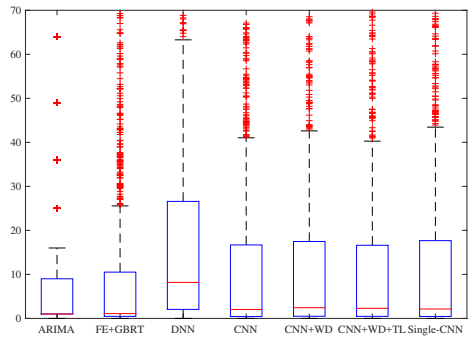
(a) Region1



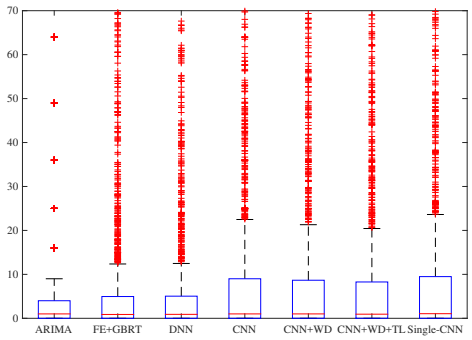
(b) Region2



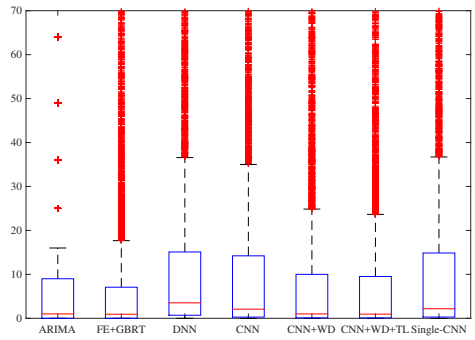
(c) Region3



(d) Region4



(e) Region5



(f) All regions

图 4.4 所有方法在 5 个区域内测试数据集上的箱型图

4.4.4 讨论

4.4.4.1 预测区间长度

图4.5 可以帮助我们理解预测区间长度与预测难度之间的关系，它表示的是预测区间长度变化时，平均销量均方差的变化情况。可以看到，预测区间越长则更容易预测，因为预测区间越长该预测区间上的平均销量越稳定。但是，更短的预测区间意味着更灵活的商业决策，因此在实际应用中我们需要在灵活性与准确性之间取得一个良好的平衡。

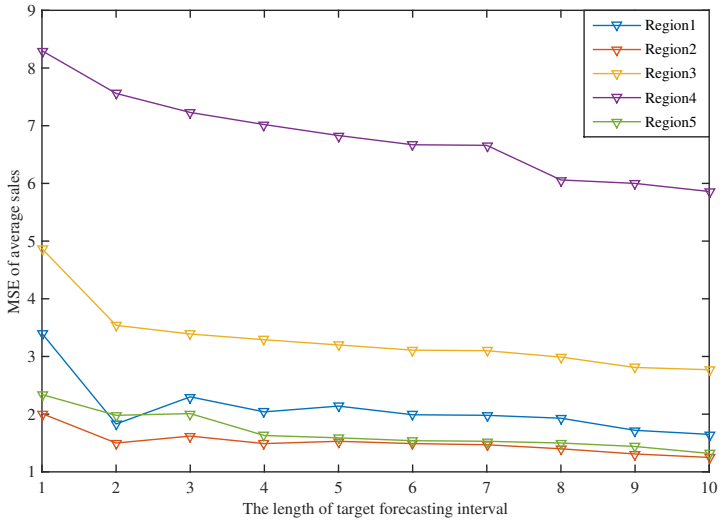


图 4.5 预测区间长度变化时，平均销量均方差的变化情况

4.4.4.2 数据框长度

模型所使用的数据框长度是模型中重要超参数之一，它决定了模型使用多少历史数据来预测未来的销量。从图 4.6中可以看到，虽然模型对这一参数较为鲁棒，但如果数据框太短则其所包含的信息不够充分，预测效果较差；如果数据框太长则其所包含的无用信息太多，预测效果也会较差。另外，更长的数据框需要

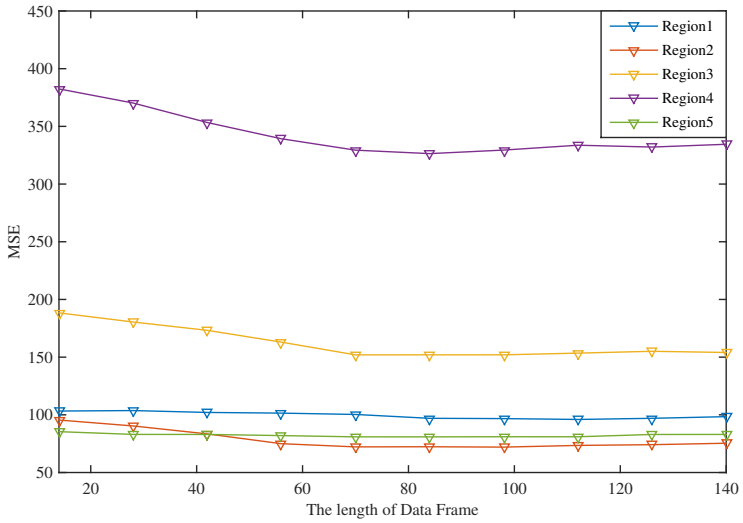


图 4.6 随着数据框长度变化，MSE 分数的变化情况

更多的计算资源。综上，在实际应用中需要在保证准确性的前提下，选取尽可能短的数据框。

4.4.4.3 权重衰减系数

我们通过公式（4.10）为训练样本赋予权重：离预测区间越近的点权重越大。公式中参数 β 用于调节样本权重衰减速度，如果 β 较大则模型更偏向于较近样本所反映的模式，反之模型则更均衡地考虑所有样本。换而言之，如果 β 较大则模型为近短期模式规律建模，反之模型则为长期模式规律建模。从图4.7可以看出，$\beta = 0.02$ 是个良好的折中，能够在考虑近短期模式规律与长期模式规律之间取得一个良好的平衡。

4.5 本章小结

在本章中，我们提出了一个新颖的方法利用卷积神经网络从结构化时序数据中自动化地提取有效特征，它可以有效地避免人工特征工程，而人工特征工程往

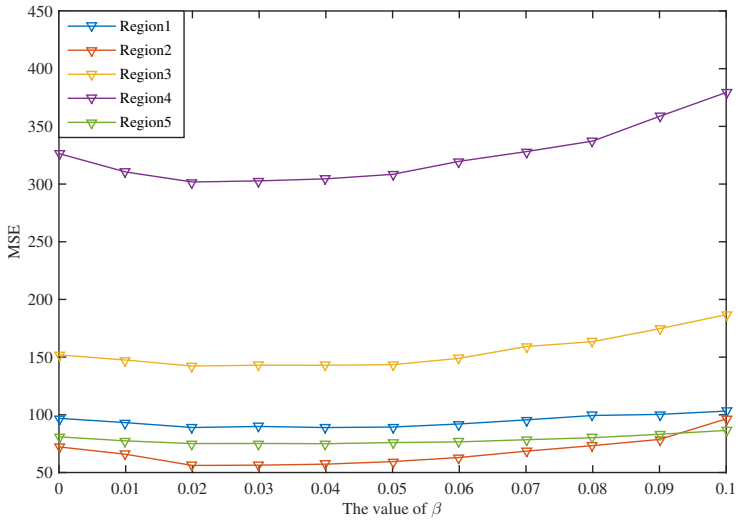


图 4.7　权重衰减中参数 β 变化时，MSE 的变化情况

往耗时、耗力且要求进行特征工程的人员具备领域知识。我们利用该方法进行商品销量预测，以商品属性信息及相关的原始日志数据为输入，以未来一段时间内商品的总销量为输出，几乎不需要任何人工干涉。首先，我们结合商品的属性信息把与之相关的原始日志数据转换为特定的“数据框”格式；然后，在该数据框上应用卷积神经网络提取有效特征；最后，在卷积神经网络的最后一层利用以这些特征为输入的线性回归预测商品销量。此外，我们还利用样本权重衰减与迁移学习等技术进一步提升了商品销量预测的准确性。在来自菜鸟网络的大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品销量预测的准确性。

接下来，我们还可以在以下几个方面做更深入的研究。首先，探索如何利用深度学习模型自动化地找出原始输入中哪些指标对最终的销量影响最大；其次，如果能找到一个统一的框架实现从所有类型的数据中自动化地提取有效特征，将会有非常高的实用价值。

第 5 章 总结与展望

5.1 本文工作总结

快速发展的电子商务为人们带来了极大的便利，而电商所处的商业环境相较传统商业环境具有更强的动态性与复杂性，这带来了诸多挑战。数据挖掘技术可以帮助人们更好地应对这些挑战，传统数据挖掘模型无法有效利用电商中的海量数据，它依赖于耗时、耗力的特征工程，得到的模型可扩展性差；深度学习模型可以有效利用大量数据，它可以实现自动化地从原始数据中抽取有效特征，得到的模型具有更高的可用性。在本文中，我们利用深度学习中的卷积神经网络对电商数据进行挖掘，针对商品搭配推荐与商品销量预测这两个方面，设计了一系列有效的算法及优化方法。本文的主要研究内容总结如下：

首先，我们设计了一个基于商品标题的全新商品搭配推荐算法，用对拍卷积神经网络对两个商品标题所组成的短文本对建模，将文本信息从原始的符号空间映射到特定的样式空间，进而在样式空间中计算两个商品间的搭配程度。在来自淘宝与亚马逊的两个大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品搭配生成质量。具体而言，其创新点包括：

- 创新性地使用商品标题为商品间的搭配关系建模。我们观察到商家为了商品更容易被用户通过搜索引擎访问，会把商品所有的重要属性信息放在标题当中。因此商品标题不但包括商品的外观信息，还包括商品的类别、品牌与适合人群等信息。

- 设计了一个对拍卷积神经网络，这个对拍神经网络以两个商品标题组成的短文本对作为输入。首先，通过卷积神经网络将每个商品的标题表示为实数向量；然后，将上述实数向量映射到特定的“样式空间”之中；最后，在样式空间中计算两个商品间的搭配分数。

- 在实际的推荐应用场景中，可以利用最近邻搜索技术对所设计神经网络的对拍部分进行加速。

- 相关工作发表于AAAI Workshop on Crowdsourcing, Deep Learning and Artificial Intelligence Agents。

其次，我们设计了一个新颖的模型，它可以从原始结构化数据中通过卷积神经网络自动化地提取有效特征，并进一步利用该方法实现商品销量预测。在来自菜鸟网络的大规模数据集上的验证实验表明，我们提出的算法能够有效地提升商品销量预测的准确性。具体而言，其创新点包括：

- 提出一个新颖的方法，利用卷积神经网络从原始结构化时序数据中自动化地提取有效特征。

- 利用该方法实现商品的销量预测。首先，结合商品的固有属性信息把与商品相关的原始日志数据转换为特定的“数据框”格式；然后，在数据框上应用卷积神经网络提取有效的特征；最后，在卷积神经网络的最后一层利用这些特征预测商品销量。

- 利用样本权重衰减与迁移学习等技术进一步提升商品销量预测的准确性。

5.2 未来工作展望

除了基于卷积神经网络的电商数据挖掘技术，基于深度学习的数据挖掘技术还可以推广到很多实际应用中，尤为让人感兴趣的是那些符合端到端形式的模型。这些方法最大的优点是免去了人工特征工程这个步骤：以原始数据作为输入，以最终目标作为输出，无需太多人工干涉，所建立的模型有更强的可用性。基于深度学习的数据挖掘技术有很大的探索空间，针对本文提出的算法有以下几个方面值得继续做深入的研究：

首先，基于商品标题的商品搭配推荐算法虽然能够较好地提升商品搭配生成质量，但模型中仍有如下几点值得探索：

- 更为复杂的文本模型可能可以进一步提升搭配效果，比如递归神经网络（recurrent neural network）中经典的 LSTM（Long Short-Term Memory）[50] 模型也经常用于文本数据的建模。
- 同时利用商品的图片和标题信息将会非常有趣，比如将图片与标题信息结合到一起形成多视角模型（multi-view model），又如将图片与标题映射到相同空间之后，实现图片与标题间的跨媒体（cross-media）检索与搭配等。

其次，基于从原始结构化数据中自动化地提取有效特征来预测商品销量虽然能够较好地提升商品销量预测的准确性，但模型中仍有如下几点值得探索：

- 利用深度学习模型自动化地找出原始输入中哪些指标对最终销量影响最大及其之间的关系将会非常有意义，深度学习既能挖掘出线性关系也能挖掘出非线性关系。找出这些指标及其与销量之间的关系将会对商业运作非常有帮助——商家可以通过改变这些指标来提升商品的销量。
- 如果能找到一个统一的框架实现从所有类型的数据中自动化地提取有效特征，将会有非常高的实用价值。这一框架不但可以帮助数据挖掘从业人员为新问题、新场景快速建模，还能减少他们对特定领域专业知识的依赖。

参考文献

[1] Pedro Domingos. A few useful things to know about machine learning[J]. Communications of the ACM, 2012, 55(10):78–87.

[2] Yoshua Bengio, Aaron Courville, Pascal Vincent. Representation learning: A review and new perspectives[J]. IEEE transactions on pattern analysis and machine intelligence, 2013, 35(8):1798–1828.

[3] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278–2324.

[4] Jiawei Han, Jian Pei, Yiwen Yin. Mining frequent patterns without candidate generation[C]. In ACM SIGMOD Record. ACM, 2000, 29, 1–12.

[5] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, David M Pennock. Methods and metrics for cold-start recommendations[C]. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2002, 253–260.

[6] Julian McAuley, Christopher Targett, Qinfeng Shi, Anton van den Hengel. Image-based recommendations on styles and substitutes[C]. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015, 43–52.

[7] Wan-I Lee, Bih-Yaw Shih, Chen-Yuan Chen. A hybrid artificial intelligence sales-forecasting system in the convenience store industry[J]. Human Factors and Ergonomics in Manufacturing & Service Industries, 2012, 22(3):188–196.

[8] Samaneh Beheshti-Kashi, Hamid Reza Karimi, Klaus-Dieter Thoben, Michael

Lütjen, Michael Teucke. A survey on retail sales forecasting and prediction in fashion markets[J]. Systems Science & Control Engineering, 2015, 3(1):154–161.

[9] Alex Krizhevsky, Ilya Sutskever, Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks[C]. In Advances in neural information processing systems. 2012, 1097–1105.

[10] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search[J]. Nature, 2016, 529(7587):484–489.

[11] Michael J Pazzani, Daniel Billsus. Content-based recommendation systems. In The adaptive web, Springer, 2007, 325–341.

[12] Vignesh Jagadeesh, Robinson Piramuthu, Anurag Bhardwaj, Wei Di, Neel Sundaresan. Large scale visual recommendations from street fashion images[C]. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014, 1925–1934.

[13] Kazuhiro Yamaguchi, Mohammad Hadi Kiapour, Tamara Berg. Paper doll parsing: Retrieving similar styles to parse clothing items[C]. In Computer Vision (ICCV), 2013 IEEE International Conference on. IEEE, 2013, 3519–3526.

[14] Wei Di, Catherine Wah, Arpit Bhardwaj, Robinson Piramuthu, Neel Sundaresan. Style finder: Fine-grained clothing style detection and retrieval[C]. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on. IEEE, 2013, 8–13.

[15] M Hadi Kiapour, Kota Yamaguchi, Alexander C Berg, Tamara L Berg. Hipster wars: Discovering elements of fashion styles. In Computer Vision–ECCV 2014, Springer, 2014, 472–488.

[16] Andreas Veit, Balazs Kovacs, Sean Bell, Julian McAuley, Kavita Bala,

Serge Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences[C]. In Proceedings of the IEEE International Conference on Computer Vision. 2015, 4642–4650.

[17] Gerald Keller, Nicoleta Gaciu. Managerial statistics[M]. South-Western Cengage Learning, 2012.

[18] William Wu-Shyong Wei. Time series analysis[M]. Addison-Wesley publ Reading, 1994.

[19] Gauri Kulkarni, PK Kannan, Wendy Moe. Using online search data to forecast new product sales[J]. Decision Support Systems, 2012, 52(3):604–611.

[20] David Lazer, Ryan Kennedy, Gary King, Alessandro Vespignani. The parable of google flu: traps in big data analysis[J]. Science, 2014, 343(6176):1203–1205.

[21] Usha Ramanathan. Supply chain collaboration for improved forecast accuracy of promotional sales[J]. International Journal of Operations & Production Management, 2012, 32(6):676–695.

[22] Jinyoung Yeo, Sungchul Kim, Eunyee Koh, Seung-won Hwang, Nedim Lipka. Browsing2purchase: Online customer model for sales forecasting in an e-commerce site[C]. In Proceedings of the 25th International Conference Companion on World Wide Web. International World Wide Web Conferences Steering Committee, 2016, 133–134.

[23] Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep learning, 2016. Book in preparation for MIT Press.

[24] Alex Graves, Abdel-rahman Mohamed, Geoffrey Hinton. Speech recognition with deep recurrent neural networks[C]. In 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013, 6645–6649.

[25] Phil Blunsom, Edward Grefenstette, Nal Kalchbrenner, et al. A convolutional neural network for modelling sentences[C]. In Proceedings of the 52nd Annual

Meeting of the Association for Computational Linguistics. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014.

[26] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. Going deeper with convolutions[C]. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015, 1–9.

[27] Tara N Sainath, Brian Kingsbury, George Saon, Hagen Soltau, Abdel-rahman Mohamed, George Dahl, Bhuvana Ramabhadran. Deep convolutional neural networks for large-scale speech tasks[J]. Neural Networks, 2015, 64:39–48.

[28] Aliaksei Severyn, Alessandro Moschitti. Learning to rank short text pairs with convolutional deep neural networks[C]. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015, 373–382.

[29] Vinod Nair, Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines[C]. In Proceedings of the 27th International Conference on Machine Learning (ICML-10). 2010, 807–814.

[30] Mihai Surdeanu, Massimiliano Ciaramita, Hugo Zaragoza. Learning to rank answers to non-factoid questions from web collections[J]. Computational Linguistics, 2011, 37(2):351–383.

[31] Raia Hadsell, Sumit Chopra, Yann LeCun. Dimensionality reduction by learning an invariant mapping[C]. In Computer vision and pattern recognition, 2006 IEEE computer society conference on. IEEE, 2006, 2, 1735–1742.

[32] Abdessamad Echihabi, Daniel Marcu. A noisy-channel approach to question answering[C]. In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1. Association for Computational Linguistics, 2003, 16–23.

[33] Antoine Bordes, Jason Weston, Nicolas Usunier. Open question answering with weakly supervised embedding models. In Machine Learning and Knowledge Discovery in Databases, Springer, 2014, 165–180.

[34] Parikshit Ram, Alexander G Gray. Maximum inner-product search using cone trees[C]. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012, 931–939.

[35] Anshumali Shrivastava, Ping Li. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips)[C]. In Advances in Neural Information Processing Systems. 2014, 2321–2329.

[36] Fumin Shen, Wei Liu, Shaoting Zhang, Yang Yang, Heng Tao Shen. Learning binary codes for maximum inner product search[C]. In Proceedings of the IEEE International Conference on Computer Vision. 2015, 4148–4156.

[37] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting.[J]. Journal of Machine Learning Research, 2014, 15(1):1929–1958.

[38] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, Jeff Dean. Distributed representations of words and phrases and their compositionality[C]. In Advances in neural information processing systems. 2013, 3111–3119.

[39] Mohit Iyyer, Peter Enns, Jordan L Boyd-Graber, Philip Resnik. Political ideology detection using recursive neural networks.[C]. In ACL (1). 2014, 1113–1122.

[40] Diederik Kingma, Jimmy Ba. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[41] Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian J. Goodfellow, Arnaud Bergeron, Nicolas Bouchard, Yoshua Bengio. Theano: new features and speed improvements[EB/OL]. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[42] Greg Linden, Brent Smith, Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering[J]. Internet Computing, IEEE, 2003, 7(1):76–80.

[43] David M Blei, Andrew Y Ng, Michael I Jordan. Latent dirichlet allocation[J]. the Journal of machine Learning research, 2003, 3:993–1022.

[44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay. Scikit-learn: Machine learning in Python[J]. Journal of Machine Learning Research, 2011, 12:2825–2830.

[45] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, Guangquan Zhang. Transfer learning using computational intelligence: A survey[J]. Knowledge Based Systems, 2015, 80:14–23.

[46] Dennis Zill, Warren S Wright, Michael R Cullen. Advanced engineering mathematics[M]. Jones & Bartlett Learning, 2011.

[47] Jerome H Friedman. Greedy function approximation: a gradient boosting machine[J]. Annals of statistics, 2001, 1189–1232.

[48] Wes McKinney. pandas: a python data analysis library[J]. see http://pandas. pydata. org, 2015.

[49] Tianqi Chen, Carlos Guestrin. Xgboost: A scalable tree boosting system[J]. arXiv preprint arXiv:1603.02754, 2016.

[50] Sepp Hochreiter, Jürgen Schmidhuber. Long short-term memory[J]. Neural computation, 1997, 9(8):1735–1780.

攻读硕士学位期间的主要研究成果

学术论文

- **Kui Zhao**, Xia Hu, Jiajun Bu, Can Wang. Deep Style Match for Complementary Recommendation[C]. Association for the Advancement of Artificial Intelligence (AAAI), Workshop on Crowdsourcing, Deep Learning and Artificial Intelligence Agents, 2017 31th International Conference on AAAI.
- **Kui Zhao**, Yi Wang, Xia Hu, Can Wang. Effective Blog Pages Extractor for Better UGC Accessing[C]. Information Science and Control Engineering (ICISCE), 2016 3rd International Conference on IEEE.
- **Kui Zhao**, Shihan Wang, Wei Wang. Fast Image Retrieval Based on Two-dimensional Embedding[C]. International Conference on Computer Science and Electronic Technology (ICCSET), 2015 4th International Conference.
- **Kui Zhao**, Jiajun Bu, Can Wang. Sales Forecast in E-commerce using Convolutional Neural Network. Submitted.
- Can Wang, **Kui Zhao**, Bangpeng Li, Zilun Peng, Jiajun Bu. Navigation Objects Extraction for Better Content Structure Understanding. Submitted.

发明专利

- 王灿、**钊魁**、卜佳俊、陈纯。一种基于导航对象提取的无障碍网络导航方法，国家发明专利，已受理。
- 王灿、**钊魁**、卜佳俊、陈纯。基于电子商务商品标题的商品搭配算法，国家发明专利，已受理。

致谢

时光荏苒，岁月如梭。转眼间我的研究生生涯即将结束，在浙江大学 Eagle 实验室的近三年里，收获颇丰。在此期间，我不但在国际顶级会议 AAAI 的 Workshop 上发表了学术论文，还拿到了业界内具有一定影响力的两个奖项："新浪微博互动预测大赛"的优胜奖与"广东航空大数据创新大赛"的冠军。

首先，我要感谢 Eagle 实验室，感谢陈纯老师和卜佳俊老师。两位老师正确地把握实验室发展的大方向，为我们提供了良好的学习与工作环境，引导学生沿着科研兴趣做深入研究，创造了自由的发展空间和浓郁的学习氛围。

其次，我要感谢我的导师王灿老师，他在工作与生活上给予我大力支持，不但是工作上的良师，也是生活上的益友。在工作的科研学习方面，以王老师的指导为基础，我深入地学习了机器学习、深度学习与数据挖掘等方面的专业知识，为今后的学习与工作打下了坚实的基础；在生活的待人接物方面，以王老师为榜样，我亦有了长足的进步。

再次，我要感谢李邦鹏、王焱、陈佳伟、史麒豪、周晟、盛夏、沈鑫、孙丽丽等同学，我们一起学习、一起研究、一起进步。是大家的陪伴，让我的学习与生活充满了欢乐和温暖。

最后，我要感谢我的家人给予我物质和精神上的支持，在你们的关心和支持下，我顺利地完成了研究生阶段的学业。

钊魁

2017 年 1 月于浙大求是园

浙江大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得浙江大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：　　签字日期：　　年　　月　　日

学位论文版权使用授权书

本学位论文作者完全了解浙江大学有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权浙江大学可以将学位论文的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：　　导师签名：

签字日期：　　年　　月　　日　　签字日期：　　年　　月　　日