

**硕 士 学 位 论 文**

MASTER’S DISSERTATION

论文题目

作者姓名刘建成

学科专业计算机科学与技术

指导教师张大鹏 副教授

**2018年5月**

中图分类号：TP312 学校代码：10216

UDC：004.8 密级：公开

**工学硕士学位论文**

|  |  |  |
| --- | --- | --- |
| 硕士研究生 | ： | 刘建成 |
| 导师 | ： | 张大鹏 副教授 |
| 申请学位 | ： | 工学硕士 |
| 学科专业 | ： | 计算机技术 |
| 所 在 单 位 | ： | 信息科学与工程学院 |
| 答 辩 日 期 | ： | 2018年5月 |
| 授予学位单位 | ： | 燕山大学 |

A Dissertation in Computer Science and Technology

**RESEARCH OF PARALLELIZED COLLABORATIVE dEEPLEARNING**

by Liu Jiancheng

Supervisor: Associate Professor Zhang Dapeng

**Yanshan University**

May, 2018

燕山大学硕士学位论文原创性声明

本人郑重声明：此处所提交的硕士学位论文《协同深度推荐算法并行化研究》，是本人在导师指导下，在燕山大学攻读硕士学位期间独立进行研究工作所取得的成果。论文中除已注明部分外不包含他人已发表或撰写过的研究成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。本声明的法律结果将完全由本人承担。

作者签字： 日期： 年 月 日

燕山大学硕士学位论文使用授权书

《协同深度推荐算法并行化研究》系本人在燕山大学攻读硕士学位期间在导师指导下完成的硕士学位论文。本论文的研究成果归燕山大学所有，本论文的研究内容不得以其它单位的名义发表。本人完全了解燕山大学关于保存、使用学位论文的规定，同意学校保留并向有关部门送交论文的复印件和电子版本，允许论文被查阅和借阅。本人授权燕山大学，可以采用影印、缩印或其它复制手段保存论文，可以公布论文的全部或部分内容。

保密□，在 年解密后适用本授权书。

本学位论文属于

不保密□。

(请在以上相应方框内打“√”)

作者签名： 日期： 年 月 日

导师签名： 日期： 年 月 日

摘 要

关键词：深度学习；推荐系统；协同深度学习；Spark集群

# Abstract

Collaborative Deep Learning relieves the problem that Conventional CF-based methods use the ratings given to items by users as the sole source of information for learning to make recommendation, while CF-based methods to degrade significantly in their recommendation performance because of the ratings are very sparse. CDL use the ability of Neural network that learn the features from data and to fit a robust model automatically. It introduces a new way to solve the problems that the recommendation system meets the very sparse data. Meanwhile, Recommendation System can not hold the Big Data because of the Rating Matrix is too large to load in the memory and the process of training for the content model is hard to predict. As a result, the Distribute Platform for RS is more and more necessary.

To solve the problem mentioned above, We propose in this paper a model based on CDL called "CDL with Item private Node"(CDL-i), And train CDL-i with a parallel method on Apache Spark. When training the CDL-i model, more than one dataset was used. we transfer CDL-i to Spark to investigate it's scalability.

First of this paper, we introduce the theory of the Parallelized CDL-i, including CDL, Autoencoder, PMF, state of parallelized machine learning and Apache Spark.

Then, we develop a modified collaborative deep learning CDL-i. Meanwhile, we introduce a method to show that how to transfer CDL-i to Spark. By the way, a framework based Spark for training machine learning model with pipeline Logic was developed.

At last, extensive experiments on real-world datasets show that our modified CDL-i can improve the performance of original CDL. And Parallel experiments can tell us the parallelized CDL was scalable and effective. As the result, CDL-i can perform a better precision, and has the ability to hold the data which has a huge size. The framework helps the algorithm easy to use.

Keywords: Deep Learning; Recommendation; CDL; Spark Cluster

目 录

[摘 要 I](#_Toc507963666)

[Abstract II](#_Toc507963667)

[第 1 章 绪 论 1](#_Toc507963668)

[1.1 课题背景及研究意义 1](#_Toc507963669)

[1.2 国内外研究现状 2](#_Toc507963670)

[1.3 现有算法存在的问题 3](#_Toc507963671)

[1.4 论文研究内容 5](#_Toc507963672)

[1.5 论文的组织结构 5](#_Toc507963673)

[第 2 章 协同深度推荐相关理论研究 7](#_Toc507963674)

[2.1 协同过滤推荐算法 7](#_Toc507963676)

[2.2 卷积神经网络 11](#_Toc507963677)

[2.3 Word embedding 13](#_Toc507963678)

[2.4 推荐系统评估方法 14](#_Toc507963679)

[2.5 本章小结 16](#_Toc507963680)

[第 3 章 协同深度推荐的模型改进 17](#_Toc507963681)

[3.1 引言 17](#_Toc507963683)

[3.2 ConvMF模型介绍 17](#_Toc507963684)

[3.2.1 ConvMF的概率模型 18](#_Toc507963685)

[3.2.2 概率矩阵分解的概率图模型 19](#_Toc507963686)

[3.3 ConvMF模型改进 21](#_Toc507963687)

[3.3.1 实验环境 23](#_Toc507963688)

[3.4 本章小结 23](#_Toc507963689)

[第 4 章 协同深度推荐模型的并行化 24](#_Toc507963690)

[4.1 引言 24](#_Toc507963692)

[4.2 数据并行 24](#_Toc507963693)

[4.3 模型并行 24](#_Toc507963694)

[4.4 协同深度推荐(CDL-i)并行 24](#_Toc507963695)

[4.5 本章小结 24](#_Toc507963696)

[第 5 章 实验及结果分析 25](#_Toc507963697)

[5.1 实验环境及软件 25](#_Toc507963699)

[5.1.1实验环境 25](#_Toc507963700)

[5.2.2实验软件 25](#_Toc507963701)

[5.2 推荐质量评价方法 25](#_Toc507963702)

[5.3 实验数据 25](#_Toc507963703)

[5.4 实验步骤 25](#_Toc507963704)

[5.5 实验分析 25](#_Toc507963705)

[5.6 本章小结 25](#_Toc507963706)

[结 论 26](#_Toc507963707)

[参考文献 27](#_Toc507963708)

[攻读硕士学位期间承担的科研任务与主要成果 28](#_Toc507963709)

[致 谢 29](#_Toc507963710)

[作者简介 30](#_Toc507963711)

1. 绪 论

## 课题背景及研究意义

近年来，互联网飞速发展，数据信息也爆发式的增长，大量的数据信息在满足人们需求的同时也造成了严重的“信息过载”(information Overload)问题。推荐系统根据用户的需求，帮助人们从信息的海洋中筛选出有用的信息，有效的解决了“信息过载”问题。目前，不同类型的推荐系统已经应用于许多领域，影响着人们生活的方方面面。比如，阿里巴巴和亚马逊将推荐系统应用于电子商务领域为用户推荐其感兴趣的商品；Facebook、微博、腾讯等将推荐系统应用于社交网络；网易音乐根据用户的口味给用户推荐其可能感兴趣的歌曲。

推荐算法作为推荐系统的核心，主要包括协调过滤、基于内容的推荐和混合推荐算法。其中，协同过滤是最经典也是应用最广泛的算法。过去几年，推荐系统在各行各业都取得了很大的成果，特别是在电子商务领域。Amazon使用了推荐技术来为其用户进行商品推荐，并由此带来了35%的销售额增长。电子商务服务中的用户和项目数量的爆炸式增长增加了用户对项目评分数据的稀疏性。最终，这种稀疏性降低了传统协同过滤技术的评分预测精度。为了提高准确度，一些推荐技术提出不仅要考虑评分信息，还要考虑用户的人口统计信息、社会网络和项目描述文件等辅助信息。

2006年Hinton等人提出了深度学习的概念。很快深度学习成为机器学习领域一个热门的研究方向。深度学习的概念来源于人工神经网络的研究。含多隐层的多层感知器就是一种深度学习结构。深度学习通过组合低层特征形成更加抽象的高层表示属性类别或特征，以发现数据的分布式特征表示。近年来，深度学习在图像处理、语音识别和自然语言处理等方面取得了革命性的进展。

将深度学习应用于推荐系统，一方面，深度学习可以通过学习用户和项目相关的海量数据从样本中学习数据集的本质特征，获取用户和项目的深层次表示。另一方面，深度学习通过从多远异构数据中进行自特征学习，将不同数据映射到一个相同的隐空间能够获得数据的同一表征，在此基础上融合传统推荐方法进行推荐，能够有效利用多源异构数据，缓解数据稀疏、冷启动和预测精度下降的问题。

以往的工作证明，深度学习在推荐系统中学习到的特征具有代表性，且对于解决推荐系统中数据稀疏问题具有有效性，可以提高学习性能。实际应用中，推荐系统的数据是复杂且稀疏的，基于矩阵分解的协同过滤方法很难学习到有效的特征信息。目前基于深度学习的推荐算法还比较少，但是初步的论文成果已经证明它们在实际数据应用中的有效性。因此，研究基于深度学习的推荐算法将是一个具有现实意义的问题。

## 国内外研究现状

目前，对推荐算法的研究越来越广泛。根据推荐算法所使用的信息和知识来源，推荐算法被分成很多类别。主流的推荐算法被分成下面五个类别：基于内容的推荐算法，基于协同过滤(Collaborative filtering, 简称CF)推荐算法，基于知识的推荐算法，基于人口统计学的推荐算法和混合推荐算法。其中协同过滤是当下最流行的推荐方法之一，该技术在不同的方向上都引起了研究者的广泛关注。

协同过滤推荐算法[Resnick 1994, Su 2009]寻找与目标用户有相似兴趣的用户群体，再根据这些用户过去喜欢过的商品对目标用户进行推荐，也就是说用户获得的推荐是基于和他有着相似兴趣爱好的人群。协同过滤的原理是，根据用户对商品或者信息的喜好，发现商品或者内容本身的相关性，或者发现用户的相关性，然后基于这些关联性进行推荐。基于协同过滤的推荐算法可以分为三个子类：基于用户的推荐(User-based CF Recommendation)，基于商品的推荐(Item-based CF Recommendation)和基于模型的推荐(Model-based CF Recommendation)。

基于用户的协同过滤算法[Zhao 2010]的基本原理是，根据用户对物品或信息的偏好，发现与当前用户口味和兴趣相似的“邻居”用户群(一般采用k-紧邻算法)，然后，基于这K个近邻的历史偏好信息，为当前用户进行推荐。基于商品的协同过滤算法[Sarwar 2001, Barragáns-Martínez 2010]的基本原理与此类似，它使用所有用户对商品或信息的偏好，发现商品和商品之间的相似度，然后根据用户的历史偏好信息，将类似的商品推荐给用户。基于模型的协同过滤算法[Breese 1998]是基于用户样本的用户喜好信息，通过模型训练(如贝叶斯模型，聚类模型或依赖网络模型)来学习用户的兴趣偏好，由此来进行相似商品的推荐。

在各种不同的协同过滤技术中，矩阵分解(Matrix factorization，简称MF)是基于模型的协同过滤中最热门的一种。矩阵分解方法通过用户-项目评分矩阵分解成一个用户相关的低秩矩阵(用户特征矩阵)和一个商品相关的低秩矩阵(商品特征矩阵)，并使用分解后的两个低秩矩阵的乘积来拟合原评分矩阵，同时对没有评分的用户-项目进行预测。一些改进工作已经被应用到该方法上，如带权重的矩阵分解方法[Nathan 2003]、基于非负矩阵的矩阵分解方法[Lee 2011]，和基于矩阵局部性的矩阵分解方法[Joonseok 2013]等，这些方法都展示了其在推荐系统上的效果。

近些年来，深度学习技术已经被证明在自然语言处理、计算机视觉和语音识别中能够学习到很好的数据特征表示。深度学习是机器学习近年来关注度非常热的研究领域，是由多个层次或多个非线性信息处理模块组成的模型，是一种监督和非监督学习方法在深度模型的更高抽象层次里的特征表示。卷积神经网络(CNN)是一种深度判别模型。其基本结构是，每个模块都由一个卷积层和一个采样层构成。这些模块通常相互堆叠，一个模块在另一个模块的上方，这样形成了一个深度模型。卷积层的权值共享和采样层对卷积层输出进行子采样，有效减少了下一层的数据率。卷积层共享的权值和采样层的池化过程，使得CNN有一些抗扭曲的能力，即有一定的不变性。目前，CNN已经在计算机视觉和图像识别[LeCun 1998; Ciresan 2012; Le 2012]应用中取得非常好的结果。近年来，CNN在自然语言处理方向也取得了很好地发展[Abdel-Hamid 2012,2013; Sainath 2013; Deng 2013]

将深度学习技术应用于推荐系统中进行特征学习，能够学习到更具有代表性的用户商品特征。目前，已经有一些工作致力于将深度学习应用到推荐系统中。[Salakhutdinov 2007]提出了基于玻尔兹曼机的协同过滤算法，得到了比Netflix系统更好的推荐效果。最近，研究者们已经提出了基于文档建模方法(如LDA主题型和堆栈去燥自动编码器(SDAE))的方法来利用项目描述文档如评论，摘要或概要。具体来说，Wang等人提出了协作主题回归模型(CTR)，其结合了主题建模(LDA)和协同过滤的概率方法。提出了不同的CTR，将LDA整合到协同过滤中，以不同的集成方法分析项目描述文档。最近，Wang等人提出将SDAE整合到概率矩阵分解(PMF)中的协同深度学习(Collaborative deep learning，简称CDL)，从而在评分预测精度方面产生更准确的潜在模型，但是基于深度学习的推荐算法研究还非常少，不够系统。

## 现有算法存在的问题

目前，对推荐算法的研究越来越广泛。根据推荐算法所使用的信息和知识来源，推荐算法被分成多种类别。主流的推荐算法主要被分成下面五个类别：基于内容的推荐算法，协同过滤推荐算法，基于知识的推荐算法，基于人口统计学的推荐算法和混合推荐算法。

1. 基于内容的推荐算法[Aggarwal 2002]旨在将和用户过去喜欢过的商品相似的商品推荐给用户。根据用户画像，能够找打最适合用户的推荐[Pazzani 2007]。这里建立用户画像的方法，可以使用基于记忆(启发式)方法，例如信息检索中的词频-逆文档频率(TF-IDF)；也可以使用基于模型的方法，例如使用决策树、贝叶斯分类器和聚类等方法得到用户的兴趣表示。基于内容的推荐算法对用户的兴趣可以很好的建模，并通过对物品属性维度的增加，获得更好的推荐精度。它的局限性在于，用户无法得到商品推荐直到他给足够的商品进行评分，即无法适应冷启动问题。另外，当用户的兴趣模型被建立起来后，系统很难再适应的改变其兴趣走向。
2. 协同过滤推荐算法[Resnick 1994, Su 2009]给用户推荐和他兴趣相似的用户过去喜欢的商品，也就是说用户获得的推荐是基于和他有着相似爱好的人群。基于协同过滤的推荐算法不需要对商品或者用户进行严格的建模，而且也不需求商品的描述是机器可理解的，因此这种方法是领域无关的，该方法很好的支持用户发现潜在的兴趣爱好。然而，由于该方法的推荐效果依赖于用户历史偏好的数据的多少盒准确性。大部分情况下，用户的历史偏好使用稀疏矩阵存储，稀疏矩阵计算上的问题会导致少部分人的错误偏好会对推荐结果的准确度有很大的影响。
3. 基于知识的推荐算法[Burke 2000]需要对商品的属性特征和用户画像(用户的兴趣特征)有一个基本的理解，该方法需要对商品和用户进行全面建模。基于知识的推荐算法利用对用户和商品的知识理解，来推导出最符合用户需求的商品。为了学习到商品和用户的知识，它需要能够获取到足够的知识且能够持续积累。该方法不会存在冷启动的问题，因为任何一个新的用户或商品进入系统时，已经有了对该用户或商品的清晰认识。它不需要的样本用户，因为推荐结果的计算不需要依靠大量的用户数据。然而该方法的不足之处在于，很难获取到足够的知识来构建用户商品知识库。
4. 基于人口统计学的推荐算法[Korfiatis 2013]是一种最易于实现的推荐算法，它只是简单的依据系统用户的基本信息发现用户的相关关系，然后将相似用户喜爱的其他物品推荐给当前用户。而这里人口统计学的基本信息包括年龄、性别、社会阶层、教育程度和地理位置等，根据这些信息可以得到用户画像。根据统计的用户画像，特定的匹配商品能够被推荐给用户。该方法通过学习用户的属性，将用户进行分类，从而给出相应的推荐。基于人口统计学的方法不需要用到用户对商品的评分，它不存在冷启动问题。然而，收集用户画像信息在一定程度上侵犯了用户的隐私，而且用户画像信息的收集可能有很大的噪音。
5. 混合推荐算法，单独使用上述推荐算法中的一种，都会存在一些优点和缺陷。现在往往将多个方法混合在一起，从而达到更好的推荐效果[Burke 2002]。关于如何组合各种推荐算法，目前有几种比较流行的组合机制。加权组合(Weighted Hybridization)用线性公式将几种不同的推荐算法按照一定的权重组合起来，具体权重值在测试集上反复试验，从而达到最好的推荐效果。切换混合(Switching Hybridization)：由于不同的情况(数据量，用户和商品等)，推荐算法的性能可能有很大的不同，该方式允许在不同的情况下，选择最为合适的推荐算法。分区混合(Mixed Hybridization)采用多种推荐算法，并将不同的推荐结果分不同的区显示给用户，该方法被很多电子商务网站采用，用户可以得到很全面的推荐，也更容易找到他们想要的东西。基于深度学习的推荐算法，也算是一种混合推荐。

## 论文研究内容

本文基于协同深度学习算法[[16](#_ENREF_16)]3其模型进行改进，提出了模型CDL-i，并进行了协同深度学习算法的并行化研究，依托Apache Spark分布式计算平台，将协同深度学习算法并行化。提高了推荐系统对数据信息的使用程度，同时提高了推荐系统在大数据时代下的可扩展性。

CDL通过联合自动编码机对项目内容的深度表示学习和协同过滤评分矩阵的反馈信息，构建了一种多层的贝叶斯模型。本文在CDL基础上针对其项目内容探测较为粗犷的不足进行了改进，提出CDL-i模型。并在真实数据集上结合多组实验证明相对于传统推荐，CDL-i带来的提升。随后本文介绍了算法的并行化改进及移植到Spark平台下的成果，同时实现了一套基于Spark分布式平台的Pipeline学习框架。

本文贡献可简要概括为：

1. 对协同深度学习进行改进，基于CDL算法提出了CDL-i模型。
2. 依托分布式计算平台Spark，对算法进行并行化改进，赋予其较强扩展性。
3. 实现并完善一套基于Spark的Pipeline分布式机器学习框架，简化机器学习工作流程。

## 论文的组织结构

本文对协同深度学习的并行化进行了研究，主要内容有：

第2章介绍课题相关理论，包括协同深度学习(CDL)、自动编码机、概率矩阵分解；机器学习算法模型并行化现状与思想；介绍了基于Spark的分布式并行技术。

第3章详细分析CDL算法，并对多种模式下的算法模型进行探索与实验对比。介绍了基于CDL的改进模型CDL-i。并通过在真实数据集上的多组实验证明改进所带来的提升。

第4章对CDL-i算法进行了并行化研究与改进，详细描述其移植到Spark分布式计算框架下实现并行训练的思想。通过对数据一系列处理，验证海量数据下并行化CDL的可扩展性，并与单机性能做了对比分析。

第5章，基于前文的理论基础与算法实现，构建了一种基于Spark平台的Pipeline分布式机器学习框架，并行版本的CDL-i算法作为其中一个算法模块被融入其中，简化了算法的应用流程。

最后作出结论与展望。CDL在推荐准确率上较大优势，改进CDL-i也为推荐性能带来提升。将算法并行化后，模型在海量数据情景下有较强扩展能力。对机器学习流程进行抽象与统一，实现的Pipeline框架提升了算法的易用性。但算法仍有改进空间，框架实现仍有些许不足。

1. 协同深度推荐相关理论研究

## 协同过滤推荐算法

推荐算法主要有基于协同过滤的推荐，基于人口统计学的推荐和基于内容的推荐等等。其中，协同过滤是当下最流行的推荐方法之一，该技术在不同的方向上都引起了研究者的广泛关注。协同过滤本质就是利用其它用户对商品的历史行为来协助对目标用户的推荐。协同过滤技术可以分成基于邻域的协同过滤方法和基于模型的协同过滤方法两种。而在不同的协同过滤技术中，矩阵分解(Matrix factorization，简称MF)是基于模型的协同过滤方法中最热门的一种。矩阵分解方法通过和用户-商品评分矩阵分解成一个用户相关的低秩矩阵(用户特征矩阵)和一个商品相关的低秩矩阵(商品特征矩阵)，并使用分解后的两个低秩矩阵的乘积来拟合原评分矩阵，同时对未观察到的用户-商品对进行预测。

基于矩阵分解的方法中，重要的因素是所学习的用户特征因子矩阵和商品特征因子矩阵，好的用户商品特征表示能带来更好的预测结果。

推荐算法具有非常多的应用场景和商业价值，因此对推荐算法值得好好研究。推荐算法种类很多，但目前应用最广泛的应该是协同过滤类别的推荐算法。

一般来说，协同过滤推荐分为三种类型。第一种是基于用户(user-based)的协同过滤，第二种是基于项目(item-based)的协同过滤，第三种是基于模型(model-based)的协同过滤。

基于用户(user-based)的协同过滤主要考虑的是用户和用户之间的相似度，只要找出相似用户喜欢的物品，并预测目标用户对对应物品的评分，就可以找到评分最高的若干个物品推荐给用户。而基于项目(item-based)的协同过滤和基于用户的协同过滤类似，只不过这时我们转向找到物品和物品之间的相似度，只有找到目标用户对某些物品的评分，那么我们就可以对相似度高的类似物品进行预测，将评分最高的若干个相似物品推荐给用户。比如你网上买了一本机器学习的书，网站上马上就会推荐一堆机器学习，大数据相关的书给你，这里就明显用到了基于项目的协同过滤的思想。

基于用户的协同过滤需要在线找用户和用户之间的相似度关系，计算复杂度肯定会比基于项目的协同过滤高。但是可以帮助用户找到新类别的有惊喜的物品。而基于项目的协同过滤，由于考虑的物品的相似性一段时间不会改变，因此可以很容易的离线计算，准确度一般也可以接受，但是推荐的多样性来说，就很难带给用户惊喜了。一般对于小型的推荐系统来说，基于项目的协同过滤肯定是主流。但是如果是大型的推荐系统来说，则可以考虑基于用户的协同过滤，当然更加可以考虑我们的第三种类型，基于模型的协同过滤。

基于模型(model based)的协同过滤是目前最主流的协同过滤类型了。用矩阵分解做协同过滤是目前使用也很广泛的一种方法。由于传统的奇异值分解SVD要求矩阵不能有缺失数据，必须是稠密的，而我们的用户物品评分矩阵是一个很典型的稀疏矩阵，直接使用传统的SVD到协同过滤是比较复杂的。

协同过滤推荐算法[Resnick 1994, Su 2009]给用户推荐和他兴趣相似的用户过去喜欢的商品，也就是说用户获得的推荐是基于和他有着相似爱好的人群。协同过滤的原理是，根据用户对商品或者信息的喜好，发现商品或者内容本身的相关性，或者发现用户的相关性，然后基于这些关联性进行推荐。基于协同过滤的推荐算法可以分成三个子类：基于用户的推荐(User-based CF Recommendation)，基于商品的推荐(Item-based CF Recommendation)和基于模型的推荐(Model-based CF Recommendation)。

基于用户的协同过滤推荐算法[Zhao 2010]的基本原理是，根据所有用户对物品或者信息的偏好，发现与当前用户口味和兴趣相似的“邻居”用户群(一般采用*k*-近邻算法)，然后，基于这K个近邻的历史偏好信息，为当前用户进行推荐。基于商品的协同过滤算法[Sarwar 2001, Barragáns-Martínez 2010]的基本原理与此类似，它使用所有用户对商品或者信息的偏好，发现商品和商品之间的相似度，然后根据用户的历史偏好信息，将类似的商品推荐给用户。基于模型的协同过滤算法[Breese 1998]是基于用户样本的用户喜好信息，通过模型训练(如贝叶斯模型，聚类模型或依赖网络模型)来学习用户的兴趣爱好，由此来进行相似商品的推荐。

1.简单介绍协同过滤推荐算法

基于协同过滤的推荐算法不需要对商品或者用户进行严格的建模，而且也不要求商品的描述是机器可理解的，因此这种方法是领域无关的，该方法很好的支持用户发现潜在的兴趣爱好。然而，由于该方法的推荐效果依赖于用户历史偏好数据的多少和准确性。大部分情况下，用户的历史偏好使用稀疏矩阵存储，稀疏矩阵计算上的问题会导致少部分人的错误偏好会对推荐结果的准确度有很大的影响。

协同过滤算法是目前最普遍使用的独立算法之一。然而，基于矩阵分解的协同过滤存在数据稀疏性和冷启动等问题，在学习用户和商品特征方面，存在一定的局限性。

2.引出基于矩阵分解PMF的推荐，并详细介绍

随着Netflix Prize推荐比赛的成功举办，近年来隐语义模型(Latent Factor Model, LFM)受到越来越多的关注。隐语义模型最早在文本挖掘领域被提出，用于寻找文本的隐含语义。矩阵分解技术是实现隐语义模型使用最为广泛的一种方法，其思想也正是来源于此，著名的推荐领域大神Yehuda Koren更是凭借矩阵分解模型勇夺Netflix Prize推荐比赛冠军，实验结果表明，在个性化推荐中使用矩阵分解模型要明显优于传统的基于邻域的协同过滤(又称基于记忆的协同过滤)方法，如UserCF、ItemCF等，这也使得矩阵分解成为了目前个性化推荐研究领域中的主流模型。

现实生活中的User-Item矩阵极大(User数量极大，Item数量极大)，而用户的兴趣和消费能力有限，对单个用户来说消费的物品，产生评分记录的物品是极少的。这样造成了User-Item矩阵含有大量的空值，数据极为稀疏。矩阵分解的核心思想认为用户的兴趣只受少数几个因素的影响，因此将稀疏且高维的User-Item评分矩阵分解为两个低维矩阵，即通过User，Item评分信息学习到的用户特征矩阵P和物品特征矩阵Q，通过重构的低维矩阵预测用户对产品的评分。

如上所述，User-Item评分矩阵维度较高且极为稀疏，传统的奇异值分解方法只能对稠密矩阵进行分解，即不允许所分解矩阵有空值。因而，若采用奇异值分解，需要首先填充User-Item评分矩阵。这样就造成了两个问题，其一，填充数据大大增加了数据量，增加了算法复杂度；其二，简单粗暴的数据填充很容易造成数据失真。这些问题导致了传统的SVD矩阵分解并不理想。

矩阵分解方法将高维User-Item评分矩阵映射为两个低维用户和物品矩阵，解决了数据稀疏性问题。使用矩阵分解具有以下优点：

1. 比较容易编程实现，随机梯度下降方法依次迭代即可训练出模型。

比较低的时间和空间复杂度，高维矩阵映射为两个低维矩阵节省了存储空间，训练过程比较费时，但是可以离线完成；评分预测一般在线计算，直接使用离线训练得到的参数，可以实时推荐。

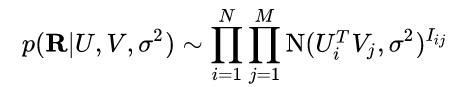
1. 预测的精度比较高，预测准确率要高于基于领域的协同过滤以及内容过滤等方法。
2. 非常好的扩展性，很方便在用户特征向量和物品特征向量中添加其它因素

许多现有的协同过滤方法既不能处理非常大的数据集，也不能轻松处理评分很的的用户。在本文中，我们采用概率矩阵分解（PMF）模型，该模型与观测值的数量呈线性关系，更重要的是，在大型，稀疏且不平衡的数据集上也有良好的表现。

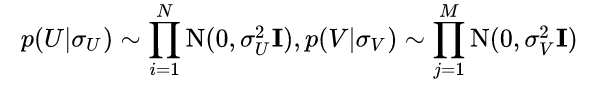
假设有N个用户和M个物品，那么就形成了一个N x M评分矩阵R，通常R非常稀疏，只有不到1%的元素是已知的，而我们要估计出缺失元素的值。PMF假设评分矩阵中的元素Ri,j是由用户的潜在偏好向量Ui和物品的潜在属性向量Vj的内积决定的，即：



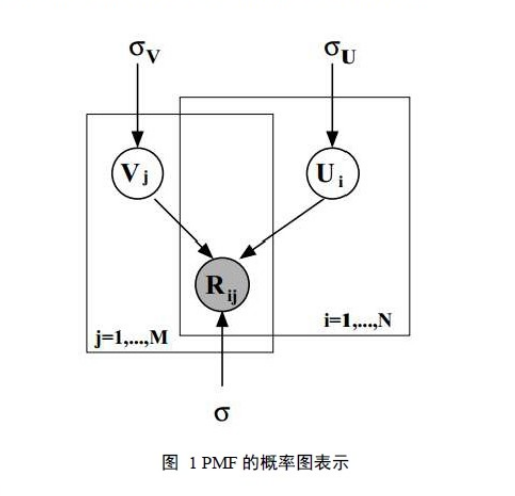
其中N表示正态分布。则观测到的评分矩阵条件概率为：



Ii,j是指示函数，若观测到Ri,j则其值为1，否则为0。再假设用户偏好向量和物品偏好向量也都服从正态分布，即：



最后，下图用概率图模型表示PMF：



## 卷积神经网络

1. 介绍神经网络的基本知识

当我们听到CNN时，一般会想到计算机视觉(computer vision)。CNN在图像分类中取得了重大的突破，也是从Facebook的图像标注到自动驾驶等，当今计算机视觉系统的核心。

虽然CNN最初是为计算机视觉开发的，但是CNN的关键思想已经积极应用于信息检索和自然语言处理(NLP)，如搜索查询检索，句子建模和分类和其他传统的NLP任务。虽然针对NLP任务需要对CNN架构进行大量修改，但最终都有助于提高各种NLP任务的性能。

传统的词袋模型或者连续词袋模型都可以通过构建一个全连接的神经网络对句子进行情感标签的分类，但是这样存在一个问题，通过激活函数可以让某些结点激活，但是由于在这样网络构建里，句子中词语的顺序被忽略，我们无法捕获由两个连续词所构成的关键特征的词的含义。在语言模型里n-gram模型将连续的两个词作为一个整体纳入到模型中，可以用来解决上面提出的问题。但是如果我们使用多元模型，实际训练时的参数时一个非常大的问题，因为假设你有20000个词，加入bi-gram实际上你就要有400000000个词，这样参数训练显然是爆炸的。另外一点，相似的词语在这样的模型中不能共享例如参数权重等，这样就会导致相似词无法获得交互信息。

现有的集成模型并不能完全捕获文档信息，因为它们使用的词袋模型忽略诸如周围单词和单词顺序的文档的上下文信息。 例如，假设在文件中给出了以下两个句子：“人们相信这个人”，“人们最终背叛了他的信任”，由于LDA和SDAE认为该文件是一个不可忽略的单词的集合，所以他们无法区分每一个事件 的“信任”。 确切地说，虽然每个“信任”的发生似乎都具有几乎相同的含义，但是这些词之间存在微妙的句法差异 - 一个动词和一个名词。 文档中的这种微妙的差异对于更深入地理解文档也是一个非常重要的因素，并且进一步的这种理解有助于提高评分预测精度。

为了解决上述问题，我们采用卷积神经网络（CNN），它是最先进的机器学习方法，在诸如计算机视觉，自然语言处理（NLP）和信息检索等各个领域表现出了高性能。为了解决上述问题，我们采用卷积神经网络（CNN），它是最先进的机器学习方法，在诸如计算机视觉，自然语言处理（NLP）和信息检索等各个领域表现出了高性能。

卷积神经网络(Convolutional Neural Network，简称 CNN)是具有以下组件的前馈神经网络的变体：1）用于产生局部特征的卷积层，2）通过仅选择几个典型的局部特征（即，通过激活函数获得具有最高分数的特征）来表示数据，作为更简洁表示的池化（或子采样）层，局部特征来自上一层，其通常是卷积层。

在图像中卷积核通常是对图像的一小块区域进行计算，而在文本中，一句话所构成的词向量作为输入。每一行代表一个词的词向量，所以在处理文本时，卷积核通常覆盖上下几行的词，所以此时卷积核的宽度与输入的宽度相同，通过这种方式，我们就能够捕捉到多个连续词之间的特征，并且能够在同一类特征计算中共享权重。卷积神经网络的一个重要概念就是池化层，一般是在卷积层之后。池化层对输入做降采用。池化的过程实际上也是对卷积层分区域求最大值或者对每个卷积层求最大值。首先，池化可以输出一个固定大小的矩阵，这对于自然语言处理当中输入句子的长度不一有非常大的作用。例如，如果你用了200个卷积核，并对每个输出使用最大池化，那么无论卷积核的尺寸是多大，也无论输入数据的维度或者单词个数如何变化，你都将得到一个200维的输出。这让你可以应对不同长度的句子和不同大小的卷积核，但总是得到一个相同维度的输出结果，用作最后的分类。另外池化层在降低数据维度的同时还能够保留显著的特征。每一种卷积核都是用来检测一种特定的特征。在以句子分类中，每一种卷积核可以用来检测某一种含义的词组，如果这种类型的含义的词语出现了，该卷积核的输出值就会非常大，通过池化过程就能够尽可能地将该信息保留下来。

1. 介绍卷积神经网络在推荐中应用(这部分可以参考原论文)

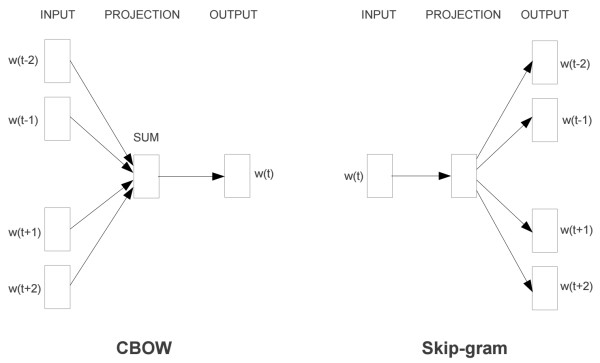
然而，CNN尚未被积极应用于推荐系统领域。据我们所知，van den Oord等人首先将CNN 应用于音乐推荐，通过声学分析观点利用CNN分析了歌曲，并提出了一种基于声学CNN获得的项目潜在模型来预测评分模型。然而，他们的CNN模型，专为声信号处理而设计，不适合处理文档。文档和声学信号对周围特征的质量有固有的差异。一定时间的信号固有地类似于其周围的信号，即具有轻微时间差的信号，而文档中某个位置处的一个字与周围的字有很大的语义差异。周围特征之间的这种相似度的差异影响局部特征的质量，最终需要不同的CNN架构。此外，该模型还没有充分反映协同信息。具体来说，项目潜在模型主要由通过CNN进行音频信号分析的结果而不是协同信息决定的。因此，总体推荐的性能甚至没有达到加权矩阵分解（WMF），WMF这是处理隐式反馈数据集的常规基于MF的协同过滤之一。

## Word embedding

Word embedding的训练方法大致可以分为两类：一类是无监督或弱监督的预训练；一类是端对端（end to end）的有监督训练。无监督或弱监督的预训练以word2vec和auto-encoder为代表。

Word2Vec是从大量文本预料中以无监督的方法学习语义知识的一种模型，它被大量地用在自然语言处理(NLP)中。Word2Vec其实就是通过学习文本用词向量的方式表征词的语义信息，即通过一个嵌入空间使得语义上相似的单词在该空间内距离很近。Embedding其实就是一个映射，将单词从原先所属的空间映射到新的多维空间中，也就是把原先词所在空间嵌入到一个新的空间中去。

Word2Vec模型中，主要有Skip-Gram和CBOW两种模型，从直观上理解，Skip-Gram是给定input word来预测上下文。而CBOW是给定上下文，来预测input word。

Word2Vec模型实际上分为了两部分，第一部分为建立模型，第二部分是通过模型获取嵌入词向量。Word2Vec的整个建模过程实际上与自动编码器(auto-encoder)的思想和相似，即先基于训练数据构建一个神经网络，当这个模型训练好以后，我们并不会用这个训练好的模型处理新的任务，我们真正需要的是这个模型通过训练数据所学得的参数，例如隐层的权重矩阵。这些权重在Word2Vec中实际上就是我们试图去学习的”word vectors”。

Word2vec和auto-encoder这类模型的特点是，不需要大量的人工标记样本就可以得到质量不错的embedding向量。不过因为缺少了任务导向，可能和我们要解决的问题还有一定的距离。因此，人们往往会在得到预训练的embedding向量后，用少量人工标注的样本去fine-tune整个模型。

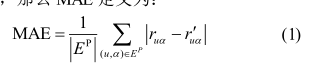
相比之下，端对端的有监督模型在最近几年里越来越受人们的关注。与无监督模型相比，端对端的模型在结构上往往更加复杂。同时，也因为有着明确的任务导向，端对端模型学习到的embedding向量往往更加准确。例如，通过一个embedding层和若干个卷积层连接而成的深度神经网络以实现对句子的情感分类，可以学习到语义更丰富的词向量表达。

## 推荐系统评估方法

面对众多的推荐算法，如何有效地评价推荐系统的优劣成为了颇具挑战的问题。[24-26]目前已有的推荐系统的评价指标数不胜数，本文只对最常用的几种进行介绍：

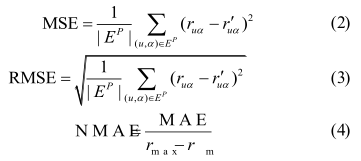
1. 预测评分的准确度

顾明思议，这类方法评价的是推荐算法所预测的用户可能评分和用户的实际评分的贴近程度。预测评分的准确度的方法目前也有很多种，其中最经典的方法是平均绝对值误差(mean absolute error,MAE)[30-32]，其定义为：

其中rua表示用户u对商品a的真实评分，rua’表示用户u对商品a的预测评分，Ep代表的是测试集。

MAE求的是单个预测评分与真实评分的差的绝对值求和然后再求平均，来表示平均绝对误差。MAE所得的值越小，说明预测评分与用户的实际评分差值越小，推荐系统的效果也越好。

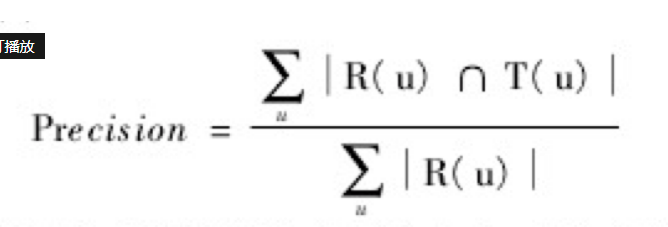
此外，平均平方误差(mean squared error,MSE)、均方根误差(root mean squared error, RMSE)以及标准平均绝对误差【15】(normalized meand absolute error,NMAE)都是和平均绝对误差类似的方法。它们分别定义为



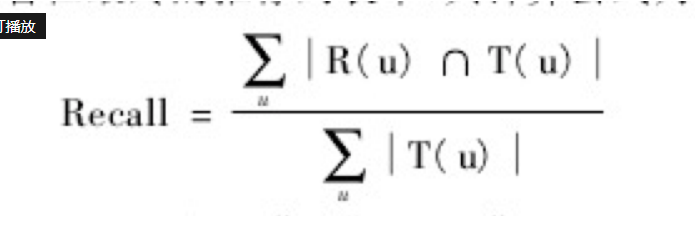
(4)式中的rmax和rmin分别为用户评分区间的最大值和最小值。由于MSE和RMSE对预测值和真实值的绝对误差做了平方，所以它们对比较大的绝对误差有更重的处罚。NMAE对结果在评分区间上做了归一化，所以可以用来评价同一推荐算法在不同数据集上的表现。

1. 分类准确度

与预测评分准确度不同，分类准确度不会要求推荐系统准确的预测用户的评分，它关注的是推荐系统能否正确预测用户喜欢或不喜欢某个商品。因此，当用户只有二元选择时，即要么喜欢，要么不喜欢，用分类准确度进行评价较为合适。目前为止，最常用的分类准确度指标有准确率(precision)、召回率(recall)、F1指标。

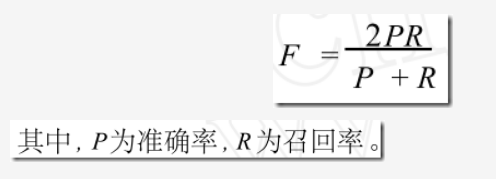
文献【38-41】最先把准确率和召回率用在了推荐系统的评价中。准确率定义为系统的推荐列表中用户喜欢的产品和所有被推荐产品的比率。准确率表示用户对一个被推荐产品感兴趣的可能性。假设R(u)表示向用户u推荐的N个物品的集合，T(u)表示数据集合中用户u真实评价过的物品的集合，则准确率的公式表示如下：

召回率定义为推荐列表中用户喜欢的产品与系统中用户喜欢的所有产品的比率，召回率表示一个用户喜欢的产品被推荐的概率。



利用准确率和召回率对推荐系统进行评价的最大问题在于它们必须要一起使用才能全面评价算法的好坏。

为了同时考察准确率和召回率，Pazzan iM[44-45]等把二者综合考虑提出了F1指标。F1指标定义为：



## 本章小结

1. 协同深度推荐的模型改进

## 引言

为了解决评分数据的稀疏性以及提高推荐系统的推荐质量，人们提出了文档建模的方法通过利用文本数据(例如评论，摘要或概要)来提高准确性。然后，由于词袋模型的固有限制，不能充分利用文档的上下文信息，对文档的理解不深刻。Kim【CONVMF】等人提出了一种新颖的文档上下文感知推荐系统，将卷积神经网络(CNN)集成到概率矩阵分解(PMF)中的卷积矩阵分解(ConvMF)。CNN通过建模组件有效地捕获图像或文档的局部特征，这些模型组件包括局部感知野，共享权重和下采样。CNN的使用可以有助于更深入的了解文档信息，产生比LDA和SDAE更好的潜在模型，特别是对于由于缺乏评分而诉诸于其描述文档的项目。本文在该模型的基础上，对CNN的Max Pooling层进行改进，使其能更充分的利用项目的描述文档，提高推荐系统的质量。

## ConvMF模型介绍

现有的CNN不适合推荐任务，因为卷积网络的目标与推荐目标不同。具体来说，常规CNN主要解决分类任务，即预测单词，短语或文档的标签。相反，推荐的目标被认为是一个回归任务，旨在准确地近似目标对项目的评分。因此，现有的CNN不能直接适用于现有的推荐任务。

ConvMF正是将CNN无缝集成到PMF中，而PMF通常用于推荐任务。因此，综合模式遵循推荐目标，最终有效地利用协作信息和上下文信息。因此，即使评分数据非常稀疏，ConvMF也能准确地预测未知的评分。

### ConvMF的概率模型

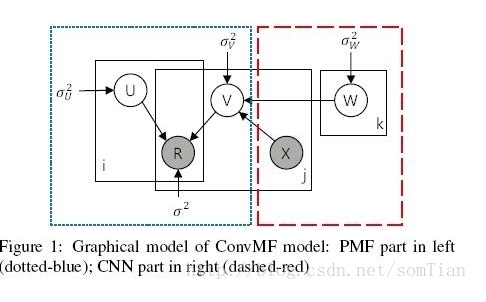


图1显示了ConvMF的概率模型的概述，它将CNN集成到PMF中。假设我们有N个用户和M个项目，并且观察到的评分由R∈RM∗N的矩阵表示。然后，我们的目标是找到其乘积(UTV)重建评级矩阵R的用户和项目潜在模型（U∈Rk∗N和V∈Rk∗M）。从概率观点上看，观察到的评分条件分布由下式给出的：

p(R|U,V,σ2)=∏Ni∏MjN(rij|uTivj,σ2)Iij

其中N（x|μ,σ2）是具有平均值μ 和方差σ2的高斯正态分布的概率密度函数，Iij是2.1节中提到的指标函数。

作为用户潜在模型的生成模型，将传统的先验，方差为σ2U的零均值球面高斯置于用户潜在模型上。

p(U|σ2U)=∏NiN(ui|0,σ2UI)

然而，与常规PMF中项目潜在模型的概率模型不同，我们假设项目潜在模型由三个变量生成：1)CNN中的内部权重W，2)表示项目j的文档Xj，以及3) ε变量作为高斯噪声，是我们能够进一步优化评级的项目潜在模型。因此，最终项目潜在模型通过以下等式获得。

vj=cnn(W,Xj)+ϵj

ϵj∼N(0,σ2VI)

对于W中的每个权重wk，我们使用先前最常用的零均值球面高斯先验：

p(W|σ2W)=∏kN(wk|0m,σ2W)

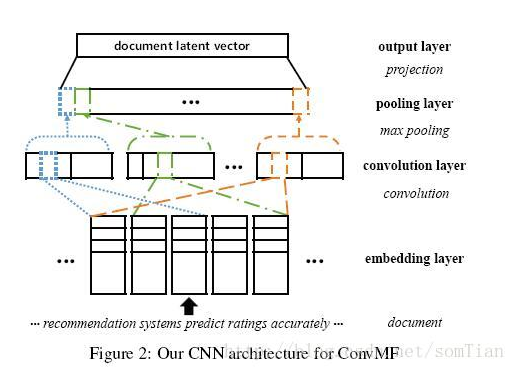
项目的隐模型的条件分布：

p(V|W,X,σ2V)=∏MjN(vj|cnn(W,Xj),σ2VI)

其中X是项目的描述文件集合。使用从CNN模型获得的文档潜在矢量作为高斯分布的平均值，并将该项目的高斯噪声用作高斯分布的方差，作为CNN和PMF之间的桥梁的重要作用，有助于完全分析两种描述文件和评分。

### 概率矩阵分解的概率图模型

本文中CNN架构的目标是从项目文档中生成文档潜在向量，用于组合具有ε变量的项目潜在模型。图2显示了该模型的CNN架构，由四层组成：1)嵌入层，2)卷积层，3)池化层，4)输出层。

Embedding Layer

嵌入层将原始文档转换成表示下一个卷积层的文档的密集数字矩阵。详细地说，关于该文件作为L个单词的序列，我们通过连接文档中单词的向量来将文档表示为矩阵。单词向量随机初始化或初始化用训练前的单词嵌入模型，如Glove。通过优化过程进一步训练单词向量。然后，文档矩阵D∈Rp∗l成为：

D=[⋅⋅⋅⋅wi−1,wi,wi+1⋅⋅⋅]

其中，l是文档的长度。P是单词Wi的维度。

Convolution Layer

卷积层提取上下文。正如我们在第二章中讨论的，文档本质上不同于信息处理或计算机视觉的上下文信息。因此，这里使用【2,11】中的卷积架构来正确分析文档。上下文特征  
cji∈R由第j个共享权重Wjc∈Rp∗ws提取，其中窗口大小ws确定周围字的数量：

cji=f(Wjc∗D(:,i:(i+ws−1))+bjc) (1)

其中 \* 是卷积运算符，bjc∈R是Wjc的偏置，f是非线性激活函数。在诸如sigmoid，tanh和整流线性单元（ReLU）的非线性激活函数中，我们使用ReLU来避免梯度消失的问题，这导致缓慢的优化收敛，并且可能导致较差的局部最小值。 然后，具有Wjc的文档的语境特征向量cj∈Rl−ws+1由等式（1）构成：

cj=[cj1,cj2,...,cji,....,cjl−ws+1] (2)

然而，一个共享权重捕获到一种类型的上下文特征。 因此，我们使用多个共享权重来捕获多种类型的上下文特征，这使得我们能够生成与Wc的数目nc一样多的上下文特征向量。

**Pooling Layer**

池化层从卷积层提取代表性的特征，并且还通过构建固定长度特征向量的池操作来处理可变长度的文档。 在卷积层之后，文档被表示为nc上下文特征向量，其中每个上下文特征向量具有可变长度（即，l−ws+1上下文特征）。 然而，这种表示方式有两个问题：1）上下文特征ci太多，其中大多数语境特征可能无助于提高性能，2）上下文特征向量的长度变化，这使得难以构建以下层。 因此，我们利用max-pooling，通过从每个上下文特征向量中提取最大上下文特征，将文档的表示形式减少为nc固定长度向量，如下所示。

df=[max(c1),max(c2),...,max(cj),...,max(cnc)]

其中cj是通过第j 个共享权重Wjc提取的长度为l−ws+1的特征。

## ConvMF模型改进

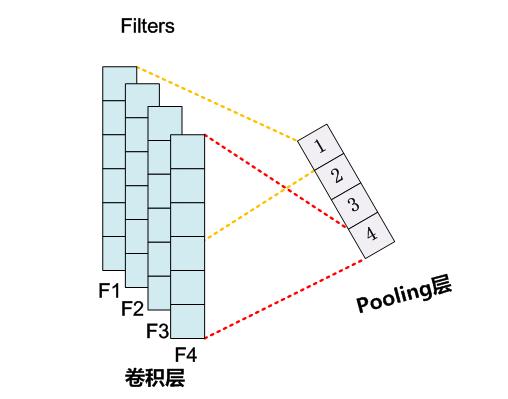
CNN是目前自然语言处理中和RNN并驾齐驱的两种最常见的深度学习模型。图1展示了在NLP任务中使用CNN模型的典型网络结构。一般而言，输入的字或者词用Word Embedding的方式表达，这样本来一维的文本信息输入就转换成了二维的输入结构，假设输入X包含m个字符，而每个字符的Word Embedding的长度为d，那么输入就是m\*d的二维向量。

这里可以看出，因为NLP中的句子长度是不同的，所以CNN的输入矩阵大小是不确定的，这取决于m的大小是多少。卷积层本质上是个特征抽取层，可以设定超参数F来指定设立多少个特征抽取器（Filter），对于某个Filter来说，可以想象有一个k\*d大小的移动窗口从输入矩阵的第一个字开始不断往后移动，其中k是Filter指定的窗口大小，d是Word Embedding长度。对于某个时刻的窗口，通过神经网络的非线性变换，将这个窗口内的输入值转换为某个特征值，随着窗口不断往后移动，这个Filter对应的特征值不断产生，形成这个Filter的特征向量。这就是卷积层抽取特征的过程。每个Filter都如此操作，形成了不同的特征抽取器。Pooling 层则对Filter的特征进行降维操作，形成最终的特征。一般在Pooling层之后连接全联接层神经网络，形成最后的分类过程。

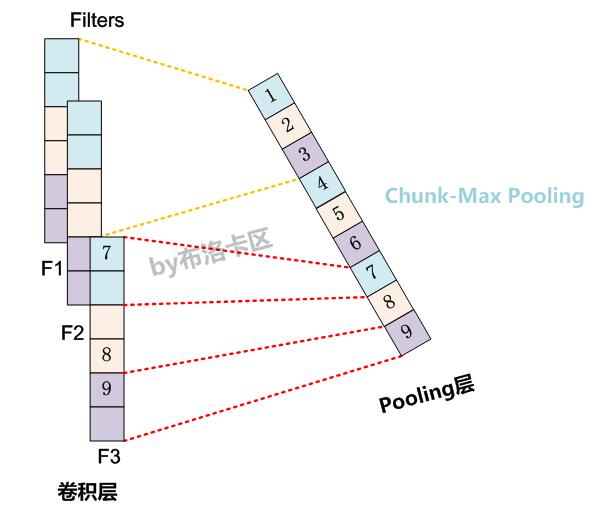
可见，卷积和Pooling是CNN中最重要的两个步骤。下面我们重点对CNN模型常见的Pooling操作方法进行改进。

MaxPooling Over Time是NLP中CNN模型中最常见的一种下采样操作。意思是对于某个Filter抽取到若干特征值，只取其中得分最大的那个值作为Pooling层保留值，其它特征值全部抛弃，值最大代表只保留这些特征中最强的，而抛弃其它弱的此类特征。

MaxPooling能减少模型参数数量，有利于减少模型过拟合问题。但是，对于文本分析任务来说，CNN模型采取MaxPooling Over Time也有很多缺点：首先在文本中，特征的出现位置是很重要的信息，比如主语出现位置一般在句子头，宾语一般出现在句子尾等等，这些位置信息其实有时候对于分类任务来说还是很重要的，但是Max Pooling基本把这些信息抛掉了。在卷积层其实是保留了特征的位置信息的，但是通过取唯一的最大值，现在Pooling层只知道这个最大值是多少，但是其出现位置信息并没有保留；另一个明显的缺点是：有时候有些强特征会出现多次，比如我们常见的TF-IDF公式，TF就是指某个特征出现的次数，出现次数越多说明这个特征越强，但是Max Pooling只保留了一个最大值，所以即使某个特征出现多次，现在也只能看到一次，就是说同一特征的轻度信息丢失了。



本文采用Chunk-MaxPooling对Max Pooling Over Time进行改进。该方法的基本思想是：把某个Filter对应的Convolution层的所有特征向量进行分段，切割成若干段后，在每个分段里面各自取得一个最大特征值，比如将某个Filter的特征向量切割成3个Chunk，那么就在每个Chunk里面取一个最大值，于是获得3个特征值。



Chunk-Max Pooling是从Convolution层先进行分段，在分段内包含特征数据里面取最大值，所以其实是一种局部Top K的特征抽取方式。至于这个Chunk怎么划分，可以有不同的做法，比如可以实现设定好段落个数，这是一种静态划分Chunk的思路；也可以根据输入的不同动态地划分Chunk间的边界位置，可以称之为动态Chunk-Max方法。

Chunk-Max Pooling很明显也是保留了多个局部Max特征值的相对顺序信息，尽管并没有保留绝对位置信息，但是因为是先划分Chunk再分别取Max值的，所以保留了比较粗粒度的模糊的位置信息；当然，如果多次出现强特征，则也可以捕获特征强度。

### 实验环境

本章所用环境为：处理器为Intel(R) Core(TM) i5-4430，GPU为3.00GHz，内存8G，硬盘150G。系统为Windows7 64位，Python版本3.5.2，Tensorflow版本为1.2.1，keras版本为2.1.1。

## 本章小结

1. 协同深度推荐模型的并行化

## 引言

## 数据并行

## 模型并行

## 协同深度推荐(CDL-i)并行

## 本章小结

1. 实验及结果分析

## 实验环境及软件

5.1.1实验环境

本文所用实验环境为：处理器为Intel(R) Core(TM) i5-4430，GPU为3.00GHz，内存8G，硬盘150G。系统为Windows7 64位，Python版本3.5.2，Tensorflow版本为1.2.1，Keras版本为2.1.1。

5.2.2实验软件

## 推荐质量评价方法

## 实验数据



## 实验步骤

## 实验分析

## 本章小结

结 论

参考文献

[1] Nagarnaik P, Thomas A. Survey on recommendation system methods[C]//International Conference on Electronics and Communication Systems, Coimbatore, India, Piscataway: IEEE, 2015:1603-1608.

攻读硕士学位期间承担的科研任务与主要成果

致 谢

转眼匆匆数年，我的研究生生涯也将要结束，人生中最美好的三年青春时光在母校燕山大学中度过，这个海滨城市中的美丽校园在我的生命中有着不同寻常的意义。行文至此，我的论文也已接近尾声，在此向在我的研究生生涯中给予过我关心、帮助和指导的各位老师和同学表示由衷的感谢！

本文的研究工作是在导师张大鹏副教授的悉心指导下完成的，论文的选题、研究和写作过程中都得到了张老师的精心指导。一直以来，张老师在学术研究、实际工作和生活学习中给予我极大的关心和鼓励。张老师深厚的理论水平、严谨的治学态度、渊博的专业知识、丰富的科研经验、敏锐的科学思维令我深感敬佩；张老师精益求精的工作作风，诲人不倦的高尚师德，严以律己、宽以待人的崇高风范，朴实无华、平易近人的人格魅力对我影响深远。正是因为有了导师严格、无私、高质量的教导，我才能在这三年的学习过程中汲取知识、提升能力。本论文从选题到完成，每一步都是在张老师的指导下完成的，本文成文倾注了张老师的心血，张老师的指导是本文成文的必要条件。同时我还要感谢计算所庄福振老师，庄老师在研究方面都给了我莫大的帮助，同时庄老师的学术态度和敬业精神也令我钦佩。本文的许多工作是在两位老师的帮助与指导下完成的。同时还要感谢王新生教授，在多方面给予我帮助。在此，谨向三位老师表示崇高的敬意和衷心的感谢！

其次，我要感谢实验室的同门师兄弟们。无论是项目方面还是课题方面，他们总是尽全力帮助我解决问题，并把全部的知识和资源分享出来，毫无保留。研究生的三年离不开他们的帮衬与照顾，在此向他们表示感谢！同时要感谢我的父母，一路走来，是家人给了我莫大的关怀与支持，可以使我可以无顾虑的求学。

随后，衷心感谢燕山大学信息科学与工程学院的全体老师的辛勤培养和谆谆教诲！感谢百忙之中评阅本论文和出席答辩的各位教授！感谢我身边的朋友、同学。同时向本领域知识产出者表示敬意！

最后，再次感谢张大鹏老师关心和付出！

作者简介

|  |  |
| --- | --- |
| D:\Documents\照片\微信图片_20180226180030.jpg | 姓名：刘建成 |
| 性别：男 |
| 民族：汉族 |
| 籍贯：河北省深州市 |
| 2011年9月考入沈阳理工大学计算机科学与技术专业。 |
| 2015年6月于沈阳理工大学取得工学学士学位。 |
| 2015年9月至今于燕山大学信息科学与工程学院攻读计算机技术专业工学硕士学位。 |