双循环迁移排序学习

赵伟强 张 熙 赖韩江 印 鉴
（中山大学数据科学与计算机学院 广州 510006）
（广东省大数据分析与处理重点实验室 广州 510006）

摘 要 迁移排序学习是信息检索领域中一个重要的研究方向，它利用带标签的源域数据来解决没有标签目标域数据的排序问题. 已有的迁移排序学习方法并没有直接解决源域与目标域的数据分布不一样的问题. 因此本文提出了一个双循环迁移排序学习方法. 该方法基于生成对抗网络，直接将源域数据映射到目标域数据中. 双循环迁移排序模型主要有两部分：(1) 让两个数据域互相学习对方域的特征分布信息部分，(2) 使用已经嵌入目标域特征信息的源域数据来训练排序模型部分. 为了更有效率，数据域特征分布学习和排序模型学习是交叉进行的，共同训练. 同时排序模型还为特征分布学习部分提供优化目标，该优化目标满足生成特征和原有标签之间的相关性，并且保持不同类别生成特征之间的距离. 在 LETOR3.0 和 LETOR4.0 标准排序学习数据集进行实验验证，结果表明本文提出的方法在平均准确率（MAP）和累计信息增益（NDCG）等评价标准下和已有主流算法相比都取得了更好的效果.

关键词 排序学习；迁移学习；生成对抗网络；信息检索；人工智能
中图法分类号 TP18 **DOI 号** 10.11897/SP.J.1016.2019.02683

Cycle Adaptation Learning to Rank

ZHAO Wei-Qiang ZHANG Xi LAI Han-Jiang YIN Jian
(*School of Data and Computer Science, Sun Yat-Sen University, Guangzhou* 510006)
(*Guangdong Key Laboratory of Big Data Analysis and Processing, Guangzhou* 510006)

Abstract Cross-domain learning-to-rank has attracted lots of attention from machine learning communities and is an important task for many practical applications. Examples include information retrieval, search engines and recommendation. The goal of cross-domain learning-to-rank is to leverage knowledge from the source domain with enough labeled data to enhance the performance of the ranking model for target domain without any labeled data. Although several ranking adaptation methods have been proposed, these existing approaches do not directly solve the problem that the source and target data are drawn from different data distributions. Such situations make the cross-domain learning-to-rank becomes a challenging problem due to the different distributions of data. To solve this problem, we propose a cycle adaptation learning-to-rank method, which is inspired by the generative adversarial networks (GANs). The motivation behind our method is to translate samples from the source domain to the target domain (and vice versa). In our cycle adaptation learning-to-rank method, two mapping functions are learned such that the distribution of data from source domain is indistinguishable from that of target domain. After these operations, the data distributions in the source domain and the target domain will be closer, which offer help to transfer the labeled information from the source domain to the target domain. More specifically,

收稿日期：2018-07-16；在线出版日期：2019-04-18. 本课题得到国家重点研发计划（2018YFB1004404）、国家自然科学基金（U1611264，U1711262，U1401256，U1501252，U1711261）资助. **赵伟强**，硕士研究生，主要研究方向为信息检索、深度学习等. E-mail：zhaowq7@mail2.sysu.edu.cn. **张 熙**，硕士研究生，主要研究方向为多媒体检索深度学习等. **赖韩江**，博士，副教授，中国计算机学会（CCF）会员，主要研究方向为信息检索、大数据分析等. **印 鉴**（通信作者），博士，教授，中国计算机学会（CCF）会员，主要研究领域为数据挖掘、大数据分析等. E-mail：issjyin@mail.sysu.edu.cn.

《计算机学报》

the pipeline of the proposed method mainly consists of two building blocks: (1) the feature distribution learning module, in which the feature distributions of two domains are learned and adapted from each other. Two generators and two discriminators are constructed in this work, in which the generators are used to generate the synthesized features that aim to let discriminators make a mistake while the discriminators learn to determine whether the feature is from true features or synthesized features. The generators and the discriminators are trained in an adversarial way. (2) The cross-domain ranking module, in which the source domain data embedded with target domain information, has been used for training the cross-domain ranking model. Since the new synthesized source data and the target data are drawn from the same data distribution, we propose to use the existing and simple pairwise methods, e. g., RankSVM and RankNet, as the basic models to train the cross-domain ranking models. In order to be more efficient, feature distribution learning and cross-domain ranking are interleaved and co-training. The target ranking model also provides an optimization objective to enhance the learning of feature distribution learning module, let the generated features satisfy the goal of ranking model, which aims to maintain the close correlation between the generated features and the ground truth label, and also preserve the distances of different generated features with different labels. Extensive experiments are conducted on the public benchmark datasets: LETOR 3.0 and LETOR 4.0. And the results show that our method demonstrates significant improvements over the several state-of-the-art baselines w. r. t MAP and NDCG two evaluation measures. The key to success comes from designing a cycle adaptation learning-to-rank to move the data distributions of two domains to be more similar. Thus our method can easily transfer the knowledge from the source domain to the target domain.

Keywords learning to rank; adaptation learning; generative adversarial networks; information retrieval; artificial intelligence

《计算机学报》

1 引 言

随着互联网技术的发展,信息数据每天都在迅速增长中.如何对大规模数据进行快速检索受到了越来越多的关注.最近,排序学习[1-8]方法被提出来,它将机器学习方法应用于训练排序模型中.给定一个查询,排序学习的目的就是把数据按照和这个查询的相关度进行排序.目前很多文献都提出了优秀排序学习算法,这些算法主要包括点级[1-2](pointwise)、对级[3-5](pairwise)和列表级[6-7](listwise)三种方法.

一般来说,要获得更好的排序学习效果需要大量专家标记的数据来进行训练.但是收集大量标记数据是一个费时费力的事情,它需要人们对每一个数据进行标注.但幸运的是,在现实中,往往会有相关领域的数据,被称为源域数据(source domain data).这些源域数据有着大量的标注信息.为了解决没有标签的目标域数据排序问题,迁移排序学习方法被提了出来.它使用已经有标签的源域数据来训练没有标签的目标域数据(target domain data)的排序模型.

在 2010 年,三种加权迁移排序学习的方法被提出[9].这些方法都是首先估计每个源域文档与目标域文档的重要性作为权重.具体来说,第一种方法是将单个文档的权重通过计算联合概率来扩展到文档对权重.加权后的文档对可以直接用于常用的排序算法之中.第二种方法探索了以跨领域查询和查询相似度作为权重的迁移学习.第三种方法结合文档对加权和查询加权两种方法,即在查询权重较大的条件下来选择权重较大的文档对.接着,在 2011 年,一种基于查询特征向量的方法被提出[10].这种方法首先聚合由一个查询中所有文档实例的查询特征向量,然后基于这种查询特征向量计算相似性后进行加权迁移学习.该方法能有效地在查询水平上估计每个查询相对于目标域的重要程度,但是也存在信息丢失的潜在危险.基于跨域加权的迁移学习方法也被提出[11],它采用了传统的 RankBoost[5]算法.学者们还研究基于多视图学习和基于列表排序方法对迁移排序学习的影响,取得了很好的效果[12].基于多视图学习是一个半监督学习框架,它旨在通过平

衡数据的不同视图之间的一致性来提高整体表现结果. 与此同时其他迁移排序学习方法也取得出色效果[13-15].

以上所有的方法都把注意力放在文档实例或查询实例的特征上，利用各种尺度来衡量加权的权重. 还没有一种方法利用两个数据域的数据分布进行迁移排序学习. 最近，对抗生成网络(Generative Adversarial Network，GAN)的研究非常热门[16-20]. 对抗生成网络的主要灵感来源于博弈论中零和博弈思想，应用到深度学习神经网络上来说，就是通过生成网络(Generator)和判别网络(Discriminator)不断博弈，进而使生成器学习到目标域数据的分布. 例如，在图像生成中，训练完成后，生成器可以从一段随机数(例如，高斯分布)生成较为逼真的图像(图像的真实分布). 在这领域中有一篇关于循环对抗生成网络的论文(Cyclegan[20])引起了广泛的关注，它能让两个数据域互相学习对方的数据分布特点，而不用具体的源数据与目标数据的关联信息.

基于此，本文提出一种双循环迁移排序算法. 该算法直接让源域和目标域互相学习对方的数据分布信息，然后利用源域学习到的目标域分布信息和源域对应的标签进行排序训练，最终用这样数据训练的排序模型应用在目标域中. 该方法基于生成对抗网络，它包括两个生成器来生成特征，两个判别器来辨别真实特征和生成特征. 同时，数据域特征分布学习和排序模型学习是共同进行的：排序模型为特征分布学习提供优化目标，特征分布为排序学习生成更适合排序的数据. 不同于已有的迁移学习算法，本文方法直接将源域数据生成到跟目标域相似的数据分布. 这样就可以将生成的数据直接应用到目标域的排序模型的学习中，解决了数据分布不同难以迁移的难题. 不同于已有的生成对抗网络，本文方法将生成器和排序模型进行交叉训练学习. 这样，生成器学习到的特征能够提高排序模型的精度，同时排序模型又能对生成器进行反馈，两者互相促进学习，解决了已有生成对抗网络只考虑数据生成而没有考虑排序的问题. 在本文中我们使用了排序学习标准数据集 LETOR3.0[21] 和 LETOR4.0[22]，与主流的迁移排序学习方法进行了对比实验，同时将本文模型跟已有的生成对抗网络进行对比，最终结果都证明本文方法比以前的方法更优.

2　相关工作

排序学习是一个热门的研究方向，已有很多的算法被提了出来. 这些算法主要可以分为三大类：点级方法、对级方法和列表级方法. 点级方法(pointwise)是将每一个样本当成一个单独输入，同时，在损失函数中也是基于一个单独的样本. 代表性的算法有 McRank[1] 和 PRank[2]，其中 PRank 是基于回归的算法，它将排序问题转换成为一个回归问题来解. McRank 则提出了将排序问题转化为多分类问题，利用多分类的算法来求解. 对级方法(pairwise)则是将数据分成一对，如(x_i, x_j)，它需要对这一对的数据进行排序，如 x_i 应该排在 x_j 前面或者相反. 经典的对级方法有 RankSVM[3]、RankNet[4] 和 RankBoost[5] 等等. RankSVM 采用了支持向量机来最小化数据对的间隔. RankNet 则采用了多层的神经网络结构来得到更好的排序函数. RankBoost 首先学习多个弱排序模型，然后用线性组合的方法将这些弱学习器合并成一个强排序模型. 列表级方法(listwise)则是将每个查询里面的所有样本当成一个输入，需要直接最小化这些样本的排序误差. 代表性的方法有 ListNet[6]，它对这些样本输出的分数定义了一个概率分布，然后使用 KL 距离来衡量跟真实分布的差距. SVM-MAP[7] 则采用了结构化 SVM 的方法来直接来优化评价标准 MAP.

上述的方法都是全监督的方法，需要大量的标签信息，但是获得大量的标签信息是一件非常费时费力的事情. 为了解决这个问题，迁移排序学习方法被提了出来，它使用了大量的、有标注信息的源域数据来提高没有标注信息的目标域数据排序模型. Chen 等人[13] 提出了跨域的排序学习模型，并分别提出了特征层面和样本层面的跨域迁移方法. Gao 等人[9] 提出了带权重的方法. 这些方法首先测量源域里面样本对目标域样本的重要度，然后将这些不同权重的源域样本迁移到目标域中. 文献[10]也是带权重的方法，不同的是，它是对每个查询进行加权而不是对每一个样本. 这也使得基于查询级实例加权的方法越来越受到重视. 文献[14]首先以目标域的每个查询为基准，在源域数据中找出和这个查询最相似的文档对，然后利用这些选出的文档对重新学习到排序模型. 文献[15]验证了不同查询级实例加权方法在公开数据集的性能. 实验表明不同的策略在不同的迁移数据中表现是不同的. 例如，基于 KL 散度的方法要好于基于分类的方法，结合文档级和查询级两种方法要比仅仅使用查询的方法要好等.

已有的方法都是使用带权重的方法，对文档或查询级进行简单的迁移，并没有直接考虑将目标域

《计算机学报》

数据分布直接转换到源域数据的分布中.最近,对抗生成网络成为了研究热点,它可以将源域数据生成到目标域数据.基于此,本文提出了基于对抗生成网络的迁移排序学习方法,它将生成网络和排序模型整合到一个端到端的网络,使得生成器可以为排序模型提供更好且有助于排序的生成数据,而且排序模型也可以指导生成器生成所需的数据.

3 双循环迁移排序

3.1 方法描述

3.1.1 问题定义

迁移学习输入的是 A 域实例特征向量 $\{\boldsymbol{x}_i^A\}_{i=1}^n$ 和对应的标签 $\{y_i\}_{i=1}^n$,以及 B 域的实例特征向量 $\{\boldsymbol{x}_j^B\}_{j=1}^m$;而输出是一个排序函数,此排序函数在 B 域中进行测试.

迁移排序学习任务是利用带标签的源域数据来解决没有标签的目标域数据的排序问题.在迁移排序学习问题中,数据域有两个:(1)带有标签的源域数据;(2)不带标签的目标域数据.本文将源域数据定义为 A,它的样本及数据域分布定义为 $\{\boldsymbol{x}_i^A\}_{i=1}^n \sim P_A$,其中 n 是源域数据样本的数量,源域数据对应的标签定义为 $Y_r=\{y_i\}_{i=1}^n$.将目标域定义为 B,同样它的数据域分布为 $\{\boldsymbol{x}_j^B\}_{j=1}^m \sim P_B$.这里的 $\boldsymbol{x}_j^B$ 为目标域第 j 个数据的特征向量.同时,本文将排序评分函数定义为 $f(\boldsymbol{x})\in\mathbb{R}^1$.输入一个数据,排序评分函数输出一个分数.要求数据越相关,评分越高,即如果 $y_i>y_j$,则 $f(\boldsymbol{x}_i^A)>f(\boldsymbol{x}_j^A)$.本文涉及的符号及定义如表 1 所示.

表 1 符号及定义

符号	定义
$\{\boldsymbol{x}_i^A\}_{i=1}^n \sim P_A, Y_r=\{y_i\}_{i=1}^n$	源域数据及其标签
$\{\boldsymbol{x}_j^B\}_{j=1}^m \sim P_B$	目标域数据
f	排序函数
G_A, G_B	生成模型,G_A将源数据生成目标域数据,G_B将目标域数据生成源域数据.
D_A, D_B	判别模型.D_A判断数据是源数据还是生成的源域数据,D_B判断数据是目标域数据还是生成的目标域数据.

3.2 模型结构和算法

在目标域中,数据是没有标签信息的,没有办法直接对目标域数据学习一个排序函数.为了解决这个问题,本文提出了双循环迁移排序算法.双循环迁移排序方法是利用 A 学习到 B 的数据分布信息从而达到迁移学习目标的方法.这种方法包含两个生成器,两个判别器和一个排序模型.判别器和生成器的角色和功能与双循环对抗生成网络相似[20],区别主要是它们的任务有所不同.在本文中它们的任务是生成便于排序的、带有目标域数据分布的数据.为了方便描述定义两个生成器分别为 G_A,G_B.两个判别器定义为 D_A,D_B.另外,排序模型算法使用 RankNet[4],RankSVM[3] 进行验证.

本文方法的具体结构图如图 1 所示,其中 G_A、G_B 的曲线代表生成过程.$\boldsymbol{x}_i^A$、$\boldsymbol{x}_i^B$ 代表数据域真实特征向量,$G_A(\boldsymbol{x}_i^A)$、$G_B(\boldsymbol{x}_i^B)$ 代表生成的特征向量.在

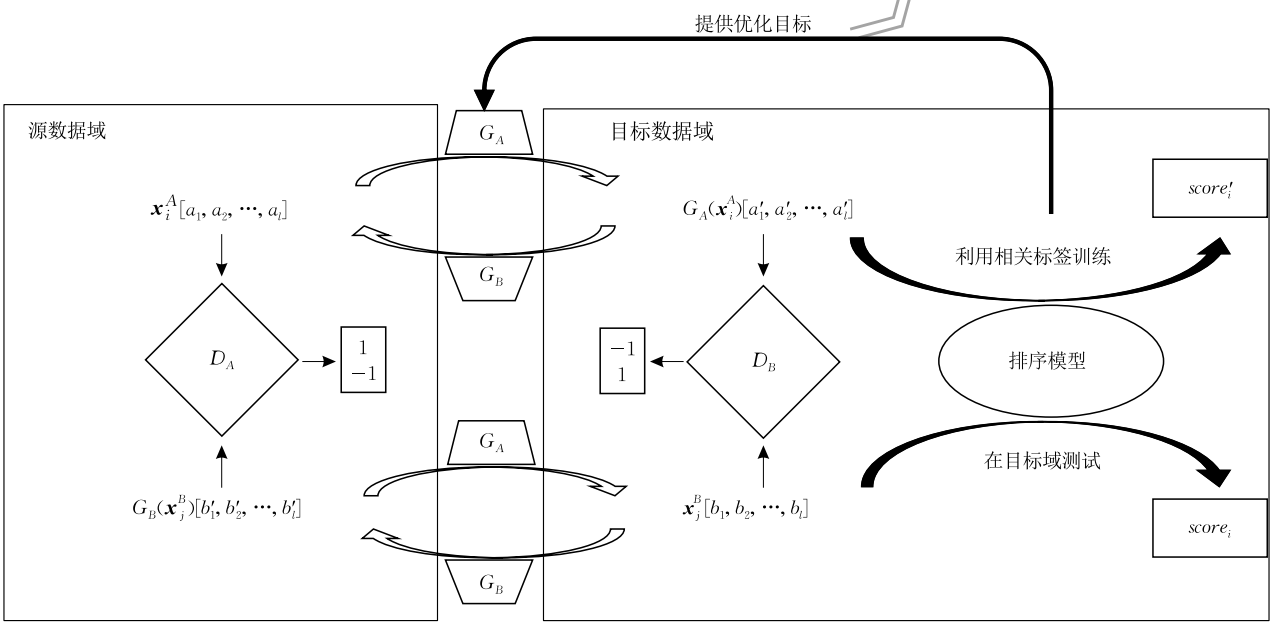

图 1 模型结构图

上面的循环中，G_A生成器将源特征域$\boldsymbol{x}_i^A$生成到目标域的$G_A(\boldsymbol{x}_i^A)$，G_B生成器再将$G_A(\boldsymbol{x}_i^A)$生成回源特征域. 下面的循环生成方向相反. D_A、D_B模块代表的判别器，用于辨别特征是各数据域的原有真实特征还是生成特征. 目标域中两条黑色曲线代表训练和测试过程，该模型用G_A生成的$G_A(\boldsymbol{x}_i^A)$和其在源数据域的标签进行训练，最终使用目标域的测试集进行测试. 在最上面的曲线代表排序模型为生成器G_A提供一个优化的目标，就是将排序模型的损失函数融入到生成器G_A的损失函数中，使生成器G_A生成出来的特征要让排序模型表现得更好. 在本文的网络中，生成器和判别器都是三层全连接网络，而排序模型和RankNet[4]的结构一致，是两层的全连接网络.

所提出方法的具体算法过程如算法 1 所示，在第 2 步中，两个生成器分别指的是G_A和G_B，同时将两个生成器生成的特征向量写入文件. 在第 3 步中读取被写入生成特征的文件，同时利用真实的特征向量训练两个判别器D_A和D_B，在这里，本文设置训练生成器训练的次数较少，所以生成的特征向量比较少，而设置判别器训练的次数较多，因此本文的模型是可以在真实数据和生成数据不成对的情况下训练，同时还能消除数据之间的相关性. 在第 4 步中，一个查询中的数据被生成器生成对应的特征向量，同时利用其原有的标签输入到排序模型中进行训练. 最后在目标域中使用f进行测试.

算法 1.　双循环迁移排序.

输入：源特征域数据，$D_r=\{\boldsymbol{x}_i^A\}_{i=1}^n$；

源特征域对应数据标签，$Y_r=\{y_i\}_{i=1}^n$；

没有标签的目标域数据，$D_t=\{\boldsymbol{x}_j^B\}_{j=1}^m$

输出：排序模型 f

1. FOR $epoch=1$；$epoch<N$；$epoch++$
2. 　用D_r和D_t中同样大小批量数据训练两个生成器；
3. 　用步骤 2 中生成的数据和各个数据域真实数据的批量数据训练两个判别器；
4. 　用批量$G_A(D_r)$数据和对应标签训练排序模型f；
5. END FOR
6. RETURN f；

3.2.1　各个部分功能描述

(1) 生成器G_A负责将$\boldsymbol{x}_i^A$生成$G_A(\boldsymbol{x}_i^A)$. 同时，它还负责将由生成器G_B生成的$G_B(\boldsymbol{x}_j^B)$生成回目标域，重构的特征向量表示为$G_A(G_B(\boldsymbol{x}_j^B))$. 这样做的目的是限制生成器$G_B$的表达空间，防止其生成的特征改变主体结构. 同样$G_A$生成的特征主体要保持在$A$域的分布，$G_B$生成的特征主体要保持在$B$域的分布. G_B的功能、角色与G_A相同，只是数据域相反.

(2) D_A负责分辨排序学习特征是属于A的$\boldsymbol{x}_i^A$和还是属于生成器G_B生成的$G_B(\boldsymbol{x}_j^B)$. 如果输入是$\boldsymbol{x}_i^A$其会判别为 1，如果输入是$G_B(\boldsymbol{x}_j^B)$它会判别为-1. D_B的功能、角色与D_A相似，但是数据域相反.

(3) 对于排序部分，由于只有源域有标签信息，所以首先使用生成器G_A和$\{\boldsymbol{x}_i^A\}_{i=1}^n$来生成$G_A(\boldsymbol{x}_i^A)$，然后排序模型使用$\{G_A(\boldsymbol{x}_i^A)\}_{i=1}^n$和$\{y_i\}_{i=1}^n$来训练一个评分函数$f$.

3.2.2　损失函数

(1) 排序模型

经典的排序问题是在给定文档特征和对应标签的情况下学习一个对文档评分的函数，其学习目标是文档与查询越相关评分越高. 本文的排序模型是被用来解决目标域B的排序问题. 由于本文使用的标准排序学习数据集，在这类数据集中，与查询相关实例$\boldsymbol{x}_i^{A+}$和不相关实例$\boldsymbol{x}_j^{A-}$组成一对，排序模型的学习目标就是$\boldsymbol{x}_i^{A+}$的评分要比$\boldsymbol{x}_j^{A-}$的评分高. 排序学习目前主要有三种方法：pointwise，pairwise 和 listwise. 本文方法主要采用的是 pairwise 方法(RankSVM[3]，RankNet[4])进行训练. 主要是因为本文方法是整合了生成模型和排序模型在一个网络结构中. 在生成模型 GAN 中，目的是从源数据分布生成到目标数据的分布. 为了达到更好的泛化性，GAN 模型的数据的输入不是整个查询. 因此，本文方法不太适合采用 listwise 方式的损失函数，因为 listwise 方法要求数据的输入是整个查询对应的文档. 另一方面，根据一些已有的工作，pairwise 的方法一般要优于 pointwise 方法[3-4]. 所以本文主要采用 pairwise 的方法来排序. 为了高效训练和保持生成后的特征与对应标签的相关性，本文还对其损失函数做出修改，即排序模型既要在目标域性能好也要在源域中性能好. 对于 RankNet 的损失函数如方程(1)：

$$L_R=-\sum_{i,j=1}^{l}\log(f(G_A(\boldsymbol{x}_i^{A+}))-f(G_A(\boldsymbol{x}_j^{A-})))-\sum_{i,j=1}^{l}\log(f(\boldsymbol{x}_i^{A+})-f(\boldsymbol{x}_j^{A-}))\tag{1}$$

其中f就是排序评分函数，$\boldsymbol{x}_i^{A+}$代表这个样本是正样本，$\boldsymbol{x}_j^{A-}$则表示这个样本是负样本. 在排序函数中，要求正样本评分要比负样本分数高，也就是评分函数值$f(\boldsymbol{x}_i^{A+})$要大于$f(\boldsymbol{x}_j^{A-})$. 对于 RankSVM 算法的损失函数如方程(2)：

$$L_R=-\sum_{i,j=1}^{l}\max\{0,1-(f(G_A(\boldsymbol{x}_i^{A+}))-f(G_A(\boldsymbol{x}_j^{A-})))\}-\sum_{i,j=1}^{l}\max\{0,1-(f(\boldsymbol{x}_i^{A+})-f(\boldsymbol{x}_j^{A-}))\}\tag{2}$$

《计算机学报》

为了达到排序目的，方程(1)和(2)需要被最小化.

(2)判别器

判别器的作用就是区分真实的排序学习特征和生成的特征.所以当判别器的输入是真实特征时它能输出1.0，而当输入是生成特征的时候它能输出−1.0.因为在Cyclegan[20]中提到，用L-2范数代替log损失函数训练模型会更为稳定，所以判别器D_A和D_B的损失函数如方程(3)、(4)：

$$L_{D_A}=E\|1.0-D_A(\boldsymbol{x}_i^A)\|^2+E\|-1.0-D_A(G_B(\boldsymbol{x}_j^B))\|^2 \tag{3}$$

$$L_{D_B}=E\|1.0-D_B(\boldsymbol{x}_j^B)\|^2+E\|-1.0-D_B(G_A(\boldsymbol{x}_i^A))\|^2 \tag{4}$$

其中，$\boldsymbol{x}_i^A$是本文前面所定义源域的真实样本数据，且$\{\boldsymbol{x}_i^A\}_{i=1}^n\sim P_A$. $\boldsymbol{x}_j^B$是本文前面所定义目标域的真实样本数据，且$\{\boldsymbol{x}_j^B\}_{i=1}^m\sim P_B$. E代表对应数据域分布条件.

(3)生成器

首先介绍循环损失函数(cycle loss)，根据Cyclegan[20]，为了限制两个生成器的表达空间，让它们能保持源特征域的主要风格和特征分布信息，在训练生成器时要使用循环损失函数，具体如方程(5)所示：

$$Cyc_Loss=E|\boldsymbol{x}_i^A-G_B(G_A(\boldsymbol{x}_i^A))|+E|\boldsymbol{x}_j^B-G_A(G_B(\boldsymbol{x}_j^B))| \tag{5}$$

简单来说，生成器G_B是用来生成特征$G_B(x)$的，同时还要将$G_A(x)$重构回源域，它的目标就是欺骗判别器D_A.所以它的损失函数可以用方程(6)来概括：

$$L_{G_B}=\lambda\times Cyc_Loss+E\|1.0-D_A(G_B(\boldsymbol{x}_j^B))\|^2 \tag{6}$$

重要的，本文为生成器G_A设计特殊的损失函数，它的目标不仅是生成$G_A(\boldsymbol{x}_i^A)$和$G_A(G_B(\boldsymbol{x}_j^B))$，而且它生成的$G_A(\boldsymbol{x}_i^A)$还要在目标域使排序模型性能好，所以在加入$Cyc_Loss$的同时也要考虑$L_R$损失函数，因为$L_R$中不仅包含用源域特征计算的损失，还能同时限制$G_A$的表达空间，不会减弱生成特征和其对应标签之间的相关性.同时为了解决生成特征的标签错乱问题，本文使用OSUDM[23]中提到的方法，具体就是保持不同类别的生成特征的距离，在本文中数据使用$G_A(\boldsymbol{x}_i^{A+})$和$G_A(\boldsymbol{x}_j^{A-})$来代表正负样本通过生成器生成出来的特征，所以距离损失函数如方程(7)所示：

$$L_{Dis}=\max\{0,|\boldsymbol{x}_i^{A+}-\boldsymbol{x}_j^{A-}|-|G_A(\boldsymbol{x}_i^{A+})-G_A(\boldsymbol{x}_j^{A-})|\} \tag{7}$$

在方程(7)中所表达的目标是生成特征的距离和源特征的距离保持不变，从而解决标签错乱问题.

至此，本文总结生成器G_A的损失函数如方程(8)所示：

$$L_{G_A}=\lambda\times Cyc_Loss+E\|1.0-D_B(G_A(\boldsymbol{x}_i^A))\|^2+\alpha\times L_R+\beta\times L_{Dis} \tag{8}$$

在本文的实验中，设置$\lambda=10.0$，$\alpha=5.0$和$\beta=5.0$.

4 实 验

4.1 数据集

为了准确地评估模型，本文首先使用排序学习标准数据集LETOR3.0[21].它包括TREC-2003和TREC-2004网页爬取的数据集.对于不同的查询任务，每个网页爬取的数据集又被分为三个不同的子集，分别是页面查找(Named Page finding，NP)、主页查找(Home Page finding，HP)和主题提炼(Topic Distillation，TD).在这些子集中有5个文件夹，分别用于五次交叉验证实验，每次交叉验证实验中数据都被分成训练集，验证集和测试集三部分.例如在训练集中，每个查询会对应一定数量的文档特征和标签.每个文档实例都是用64维的特征向量(LETOR4.0[22]中的文档实例是用46维的特征向量，OHSUMED[21]中的文档实例是45维特征向量)表示.这些特征值代表一些低水平内容特征：项目频率、反转文档等，同时也有一些高水平内容特征值：BM25、LMIR和超链接特征等等.

同时本篇文章还进行了拓展实验，一方面是为了在跨度比较大的数据分布不一致数据集上进行迁移实验，来验证本文方法的可扩展性；另一方面是与其他排序算法的比较，如ListNet，SVMMAP等.本篇论文的拓展实验使用的数据集是LETOR4.0[22]和OHSUMED[21].

LETOR4.0[22]是一个全新的排序学习标准数据集.它使用Gov2网页集(约25M页)和来自TREC 2007和TREC 2008的Million Query track的两个查询集.本文简称为MQ2007和MQ2008两个查询集.MQ2007中有大约1700个带有标记文档的查询，MQ2008中有大约800个带有标记文档的查询.

OHSUMED[21]数据集由106个查询组成，其中45个特征内容是从在线医学信息数据库MEDLINE中提取的.它包含来自270种医学期刊的348 566条记录和16 140条查询文档对，其中进行了相关性判断.

4.2 实验设置

为了公平地和以前的方法作对比，本文的实验设置遵循如下规则：

《计算机学报》

(1)所有的目标域数据的标签(包括训练集、验证集和测试集)在训练的过程中是不可见的.

(2)当进行迁移学习的时候,文件夹的顺序是一一对应保持不变的,例如源域的文件夹 1 要和目标域文件夹 1 中的数据做迁移学习. 每对迁移学习任务的结果是 5 个文件夹结果的平均值.

(3)在训练阶段不使用源特征域的验证集和测试集,而以前的方法中使用了验证集来寻找更好的参数.

(4)所有的最终测试结果都使用官方测试脚本(Eval-Score-3.0.pl,Eval-Score-4.0.pl),它能在给定与测试集一样顺序的文档实例预测分数情况下计算排序结果. 每对迁移学习任务都做五次交叉验证并取均值.

本文实现了两种排序算法,模型都使用随机梯度下降优化器进行优化. 本文的对比结果都来自以前较好的结果[9],同时为了验证本文模型端到端(End-to-End)的贡献,本文还做了使用 Cyclegan 分步训练的对比实验. 具体来说,本文主要的对比方法有:

(1) Source-only. 只使用带标签信息的源域数据进行训练. 训练好的模型用于目标域数据排序.

(2)带权重的迁移排序学习方法. Pair-weight 方法对不同的文档对进行加权;Query-weight 是对不同的查询加不同的权重,与目标域越像,权重越大;Comb-weight 结合了前两种方法,在已加权的查询下再对文档对加权;Rand-weight 则是使用了随机的加权方法.

(3) Cyclegan+ranking. 该方法也是基于生成对抗网络,它包括两部分:① 使用 Cyclegan 将源域的数据生成到目标域的分布中;② 对生成的数据进行排序. 和本文方法相比,它不是端到端的,而是分步训练.

(4) Target-only. 是用带有标签的目标域数据集训练得到的结果. 这是全监督的排序方法,由于使用了标签信息(其它迁移方法都没有使用目标域的标签信息),该方法的结果一般是最好的,它主要是用于对各种不同迁移方法的性能做一个直观的对比.

为了达到拓展实验目标,本文进行了 MQ2007 到 MQ2008 迁移实验和 MQ2008 到 OHSUMED 迁移实验,并将本文方法的实验结果与 ListNet、SVMMAP 排序学习算法做迁移实验的结果进行对比. 由于 LETOR4.0 数据集的特征维数是 46 维,OHSUMED 数据集的特征维数是 45 维,而本文算法模型需要迁移学习的两个域特征维数一致,因此,在 MQ2008 到 OHSUMED 数据集迁移学习实验前,本文利用 PCA[24] 降维技术将 MQ2008 数据集的特征维数降至 45 维.

本文中的生成器、判别器都是由三层全连接网络构成,具体网络结构如表 2 和表 3 所示,其中,Feature_size 是指所用数据的特征向量维度,例如使用 LETOR3.0 数据集 Feature_size 就是 64.

表 2　生成器网络结构设置

网络层类型	网络层内容
全连接层	120 个神经元
激活函数层	Relu
全连接层	220 个神经元
激活函数层	Relu
全连接层	Feature_size
激活函数层	Relu

表 3　判别器网络结构设置

网络层类型	网络层内容
全连接层	120 个神经元
激活函数层	Tanh
全连接层	220 个神经元
激活函数层	Tanh
全连接层	1 个神经元
激活函数层	Tanh

本文中所有的优化器都是批量随机梯度下降优化器,批量大小设置为 16.

4.3　结果及分析

本文和文献[9]设置了同样的实验数据集对,它们分别是同一年份的 HP 到 NP,NP 到 TD. 不同年份的 2003NP 到 2004TD 的迁移排序学习实验. 实验结果是由平均准确率(MAP)[25] 和累计信息增益(NDCG)[26] 作为评判标准.

对于 LETOR3.0 数据集,文档实例是由 64 维的特征向量表示的,也就是 $\boldsymbol{x}_i^A$ 和 $\boldsymbol{x}_i^B$ 是 64 维的特征向量. 因此,生成特征 $G_A(\boldsymbol{x}_i^A)$、$G_B(\boldsymbol{x}_j^B)$、$G_B(G_A(\boldsymbol{x}_i^A))$ 和 $G_A(G_B(\boldsymbol{x}_j^B))$ 都是 64 维的特征向量. 在训练生成器和判别器阶段,由于生成器和判别器是学习整个数据集数据特征的分布,所以生成特征和源域特征不是一对一的成对关系. 而在训练排序模型和测试结果的时候,本文模型对每个源域数据生成一对一对应的新特征. 例如,下面是 hp2003 的 fold1 测试集生成特征 $G_A(\boldsymbol{x}_i^A)$ 的一个样例:

0 qid:396 1:0.009804 2:0.428571 3:0.000000 4:0.333333 5:0.084577 6:0.000000 7:0.000000 … 60:0.120968 61:0.000000 62:0.000000 63:0.000000 64:0.000000 #docid=G00-34-0187122

需要注意,生成特征的标签 0 和对应的 qid 是不变的. 对于 LETOR4.0 和 OHSUMED 数据集,生成特征描述和上面相似,只是特征维度分别变为

《计算机学报》

46 维和 45 维.

4.3.1 HP 到 NP 迁移

从 HP 到 NP 数据集迁移的 MAP 结果如表 4 所示,其中第 1 行是只用源域数据训练的结果,第 2～5 行的结果来自以前较好算法的结果,第 6 行是 Cyclegan+ranking 分步训练的结果,第 7 行是本文方法的结果,第 8 行是只用目标域数据训练的结果.图 2 和图 3 是 RankSVM 排序算法下的 NDCG 结果折线图(分别是不同年份的结果图).

表 4 HP 到 NP 迁移学习的 MAP 结果

模型	RankSVM		RankNet	
	2003	2004	2003	2004
Source-only	0.417	0.596	0.497	0.610
Pair-weight	0.502	0.647	0.532	0.661
Query-weight	0.510	0.647	0.534	0.645
Comb-weight	0.542	0.665	0.562	0.679
Rand-weight	0.431	0.618	0.471	0.619
Cyclegan+ranking	0.465	0.658	0.555	0.661
本文方法	**0.626**	**0.686**	**0.634**	**0.691**
Target-only	0.660	0.681	0.672	0.669

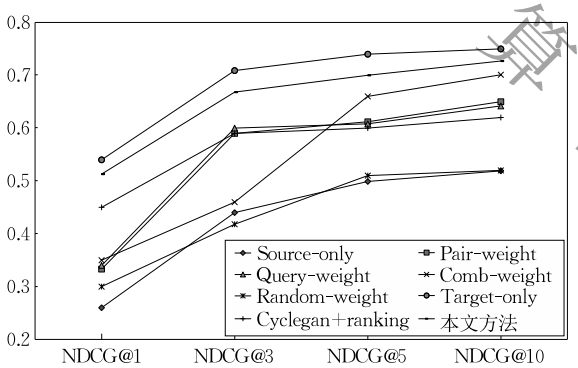

图 2 HP03 到 NP03 迁移学习的 NDCG 折线结果图

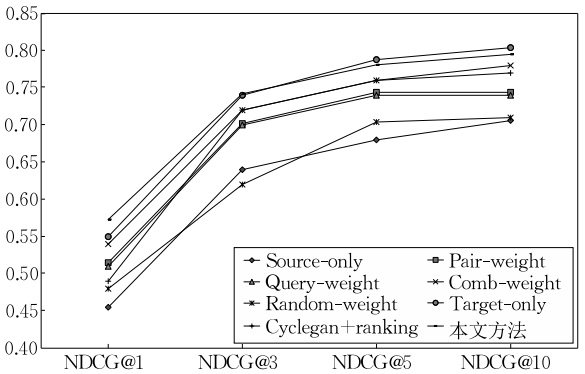

图 3 HP04 到 NP04 迁移学习的 NDCG 折线结果图

从表 4、图 2 和图 3 的结果来看,不管是 MAP 还是 NDCG 的评价标准上,本文方法都比以前的算法结果要更好.例如,在 NP03 和 HP03 的数据集来说,在 MAP 评价标准下,本文方法在 RankSVM 和 RankNet 两种不同的排序模型上性能分别比以前最好的算法提升 15.4%和 12.8%.在 NP04 和 HP04 的数据集上,本文方法分别提升 3.1%和 1.8%的性能.

从结果可以看出,源特征域通过本文的模型能很好地抓取目标特征域的分布信息,从而能够更好地将源域的数据迁移到目标域的分布中,达到更好的结果.另外,之前学者的研究[9]表明:HP 和 NP 的数据分布很接近,所以本文方法有些结果甚至比只用目标域数据训练的性能要高.

4.3.2 NP 到 TD 迁移

NP 到 TD 的迁移结果见表 5、图 4 和图 5.表 5 是 MAP 结果,图 4 和图 5 是 RankSVM 的 NDCG 对比结果图.

表 5 NP 到 TD 迁移学习的 MAP 结果

模型	RankSVM		RankNet	
	2003	2004	2003	2004
Source-only	0.146	0.176	0.196	0.164
Pair-weight	0.194	0.171	0.233	0.151
Query-weight	0.194	0.175	0.226	0.157
Comb-weight	0.222	0.179	0.235	0.158
Rand-weight	0.145	0.163	0.199	0.168
Cyclegan+ranking	0.215	0.178	0.208	0.163
本文方法	**0.231**	**0.192**	**0.225**	**0.178**
Target-only	0.235	0.205	0.266	0.180

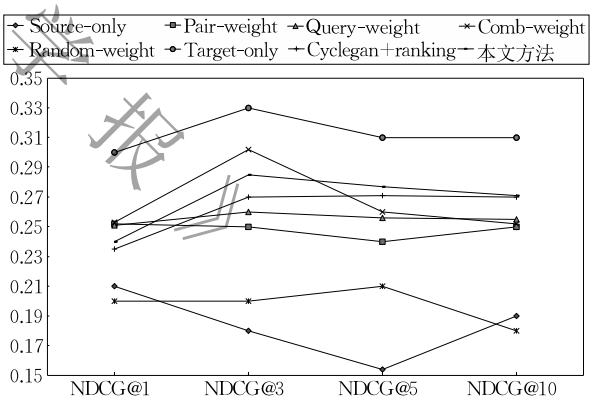

图 4 NP03 到 TD03 迁移学习的 NDCG 折线结果图

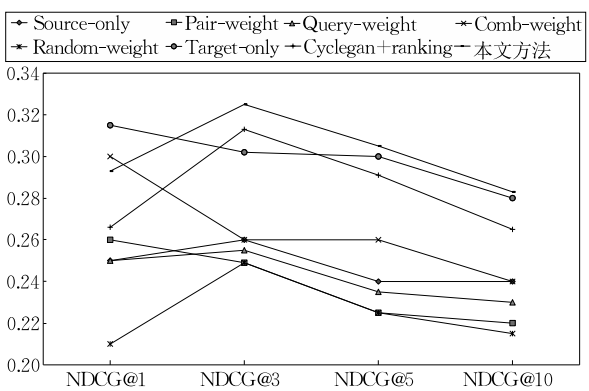

图 5 NP04 到 TD04 迁移学习的 NDCG 折线结果图

计算机学报

对于 NP03 和 TD03 数据集来说，本文的结果和以前较好的结果差不多. 经过仔细分析，目标域的特征分布信息是非常少的，两个数据集的数据量比值大约在 2∶1 左右，然而本文的方法依赖于较多的目标域特征分布信息，所以出现了此现象.

对于 NP04 和 TD04 数据集来说，由于没有上述的问题，本文方法的性能提高非常明显，与以前最好的结果比分别提升了 7.2%和 5.9%. 同分步训练的 Cyclegan＋ranking 相比，本文方法也有很大的提高. 这个主要是由于 Cyclegan＋ranking 是分步训练的，生成器与排序模型不在一起训练，导致生成的特征不能更好地用于排序模型中，这样设置实验是为了验证本文所设计损失函数的有效性. 而本文方法将生成器与排序模型一起训练，两者可以互相学习和反馈，从而性能有很大提升.

4.3.3　NP2003 到 TD2004 迁移

为了验证本文方法对不同年份不同任务的有效性，本文实现了 NP2003 到 TD2004 迁移学习对比实验. 该实验比较反映当源域数据和目标域数据分布有较大不同时各种迁移算法的效果. 表 6 是 MAP 实验结果. 本文方法与以前最好的方法相比分别提升了 13.2%和 17.1%. 本文方法使用 RankNet 做排序算法的实验结果也超越了 Target-only 方法.

表 6　NP 到 TD 迁移学习的 MAP 结果

模型	RankSVM	RankNet
	NP03-TD04	NP03-TD04
Source-only	0.145	0.136
Pair-weight	0.164	0.160
Query-weight	0.163	0.157
Comb-weight	0.166	0.164
Rand-weight	0.145	0.144
Cyclegan＋ranking	0.181	0.187
本文方法	**0.188**	**0.192**
Target-only	0.205	0.180

4.3.4　MQ 到 OHSUMED 迁移

为了验证本文提出的方法在不同大规模数据集的可扩展性、在数据分布跨度大的数据集上的性能，同时与一些更前沿算法比较，在本小节中，对 MQ2007 到 MQ2008，MQ2008 到 OHSUMED 数据集做了迁移排序学习实验. MQ2007，MQ2008 是 LETOR4.0 上面两个比较大的数据集，和 LETOR3.0 相比，它们具有更多的查询数和文档数. 这两个数据集可以更好地验证本文方法在大数据集上的效果.

另外，OHSUMED 数据集的数据分布与 MQ2008 完全不一致，MQ2008 到 OHSUMED 的迁移效果也可以让本文验证在数据分布跨度比较大的迁移效果；为了更公平地比较，排序学习的 Listwise 方法，包括 ListNet[6] 和 SVMMAP[7] 被选为对比方法. Source-only 对应的 ListNet 和 SVMMAP 模型结果表示使用相应排序学习算法在源域训练，在目标域做直接测试的实验结果；本文方法对应的结果是分别用 RankNet 和 RankSVM 为基础的双循环迁移排序学习模型的实验结果；Target-only 对应的 ListNet 和 SVMMAP 算法是在目标域训练并在目标域测试的结果. 由表 7 中可以看出，本文方法比前沿排序学习算法直接做迁移学习的结果要好，而且由于 MQ07 和 MQ08 数据集数据分布近似，所以本文模型（RankNet）甚至比 Target-only（ListNet）算法模型性能要高 2%. 但是由于降维后的 MQ08 和 OH 数据集数据分布跨度较大，使用 ListNet 和 SVMMAP 排序学习模型的结果和本文方法相差较大. 由此可见，本文方法的鲁棒性很好，并且在数据分布跨度较大的迁移排序学习任务上也能表现出色.

表 7　拓展实验迁移学习的 MAP 结果

模型	MQ07-MQ08	MQ08-OH
Source-only（ListNet）	0.452	0.388
Source-only（SVMMAP）	0.439	0.383
本文方法（RankNet）	**0.479**	**0.412**
本文方法（RankSVM）	**0.473**	**0.422**
Target-only（ListNet）	0.477	0.445
Target-only（SVMMAP）	0.456	0.445

4.3.5　领域分布程度分析

本文做了两组不同领域分布的数据迁移实验，包括了 HP 到 NP、NP 到 TD 迁移排序实验. 根据 Gao 等人[9] 对这些数据集分布的分析可知，由于这些数据集关注的是不同主题，它们的领域分布是不同的. 具体来说，TD 数据集的任务是查找涵盖主题页面的优秀网站入口点列表，HP 数据集的任务是查找个人或组织的主页，NP 数据集侧重于直接包含查询简短描述的页面. 根据 Gao 等人[9] 对数据集做 PCA 降维分析的结果可知，HP 和 NP 数据特征分布比较相近，而 TD 和它们两个分布较远. 因此，表 3、表 4 很好地代表了不同领域分布不一致的性能结果，其中表 3 中源数据与目标域的数据较接近，表 4 中 TD 和 NP 的数据分布较远.

从上面的结果可以得知：(1) 对于大部分算法来说，源数据与目标域的数据越接近，迁移的效果越好. 例如，在表 3 中，Comb-weight 使用 RankNet 在 2004 上的结果是 0.679. 这个结果比 Target-only 还要好，原因在于 HP 和 NP 这两个数据集分布接近，

计算机学报

简单的迁移方法就可以得到很好的结果. 但是在表 4 中,Comb-weight 的结果是 0.158,而 Target-only 的结果是 0.180. 由于 NP 和 TD 的分布较远,所以迁移效果就不太明显;(2) 本文提出的方法要远远好于其它方法. 例如,在表 3 的 2003 数据上,第二高的结果是 Comb-weight,它的结果是 0.542 和 0.562,而本文方法的结果是 0.626 和 0.634. 效果提升较明显. 同时,由表 4 中 NP 到 TD 的迁移结果,可以发现本文方法要比已有的方法效果更好.

在大部分情况下都是 Target-only 表现最好. 在一些数据集下,本文方法超过了 Target-only. 例如表 4 中的 2004 数据集,Comb-weight 和本文方法都比 Target-only 的方法好. 在表 7 中,MQ2007 迁移到 MQ2008 的结果中,本文方法也与 Target-only 很接近甚至高一点. 在这里主要的原因在于(1)数据分布是非常接近的. 例如文献[9]说明 HP 和 NP 数据特征分布比较相近. MQ2007 和 MQ2008 只是不同年份的数据,分布也很接近. 在这种相互接近的情况下,更容易迁移出有用的信息,可以有效地提高性能;(2) 另一个可能的原因是因为 HP2004、NP2004 和 MQ2008 数据中存在较多噪声[8]. FSMRank[8] 使用了稀疏模型,只用了小部分特征,在上面数据中都取得了更好的结果. 这也说明了为什么在 2004 上本文方法可以得到更好的结果,而在 2003 数据上没有这样的现象. 因为 2003 生成的数据中有较少的噪声,所以在结果中要比有大量噪声的结果要好.

4.3.6 数据分布可视化

为了清晰地观察学习过程中的数据分布,本文使用 TensorFlow 中的可视化工具 Tensorboard,它可以通过 Tensorflow 程序运行过程中输出的日志文件可视化 Tensorflow 程序的运行状态、数据分布等信息. Tensorboard 和 Tensorflow 程序运行在不同的进程中,Tensorboard 会自动读取最新的 Tensorflow 日志文件,并呈现当前 Tensorflow 程序运行的最新状态.

在训练模型的时候,当需要查看一个向量(在 Tensorflow 数据结构里被称为张量)在训练过程中的分布情况时,可以通过 $tf.summary.histogram$ 函数画出在随学习步数变化的数据分布直方图,这个函数是将输入的一个任意大小和形状的向量压缩成一个由宽度和数量组成的直方图数据结构. 这样,可以通过可视化向量在不同时间点的直方图来显示某些分布随时间变化的情况,如图 6、图 7 和图 8 所示. 它们是 NP2003 到 TD2004 迁移学习任务中数据分布图,其中图 6 是 $\{\boldsymbol{x}_i^A\}_{i=1}^n \sim P_A$,图 7 是生成数据分布图,即 $\{G(\boldsymbol{x}_i^A)\}_{i=1}^n \sim P_A$,图 8 是 $\{\boldsymbol{x}_j^B\}_{j=1}^m \sim P_B$ 分布示意图,其中,横轴表示值,纵轴表示数量,每个

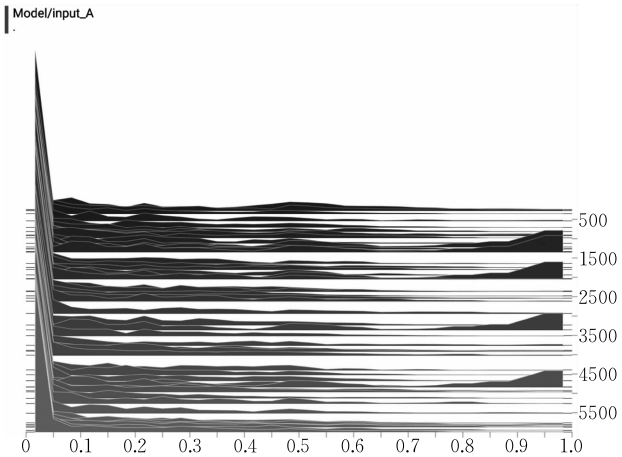

图 6 NP03 到 TD04 迁移学习的 NP03 数据分布图

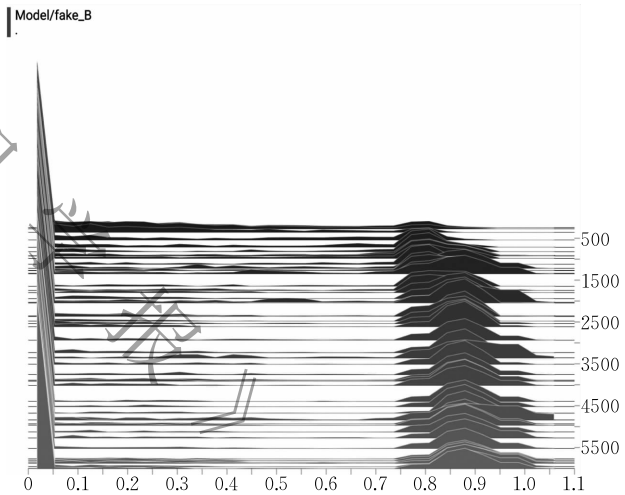

图 7 NP03 到 TD04 迁移学习的生成数据分布图

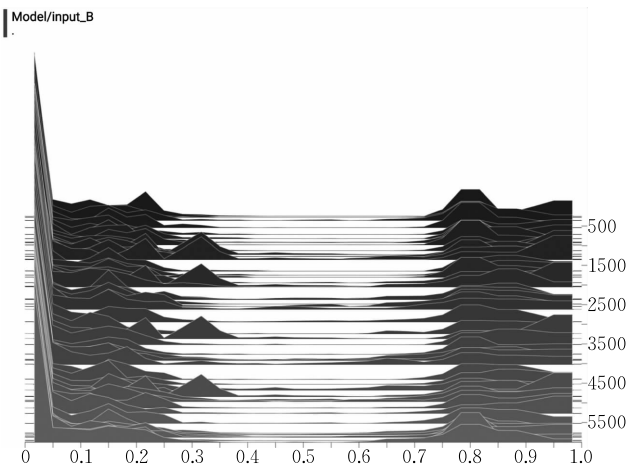

图 8 NP03 到 TD04 迁移学习的 TD04 数据分布图

切片显示一个直方图，切片按步骤（步数或者时间）排列；旧的切片比较暗，新的切片比较浅. 但是，直方图的切片并不是按每一个步数或者时间均匀分布，而是通过水塘抽样来抽取所有直方图的一个子集，以节省内存.

图中可以看出生成数据在保持原有特征基础上还学习到目标域数据分布，但其分布没有和目标域完全一样，这是因为有限制条件，这里限制条件是为了让生成特征保持与源域数据标签之间的相关性.

5 总　结

本文提出了一个双循环迁移排序学习方法. 在实际的信息检索领域可用于对没有专家标注标签的数据进行迁移学习，也可以用于提升信息检索模型的泛化能力. 本文基本的想法就是利用源数据域的数据学习目标特征域的一些数据分布信息，从而来进行排序学习，同时排序模型还要给这个数据分布信息学习的过程提供一个目标，使生成的带目标域特征信息的特征值在目标域的排序结果更好. 在学习分布信息的时候也进行一个对抗学习过程来加强分布信息的学习能力. 本文方法在 LETOR 3.0 和 LETOR4.0 标准的排序学习的数据集上进行了大量不同任务同年份不同年份等的实验和结果的对比，经过分析发现本文算法的性能更优.

由于只使用 Pairwise 的损失函数计算方法，所以本文的主题思想还可以容易结合其它更好的排序算法. 未来我们还将继续探索其他领域的前沿算法在信息检索领域的应用和发展.

致　谢　感谢审稿人提出的宝贵建议！

参　考　文　献

[1] Li P, Wu Q, Burges C J, et al. McRank: Learning to rank using multiple classification and gradient boosting//Proceedings of the 20th International Conference on Neural Information Processing Systems. Vancouver, Canada, 2007: 897-904

[2] Crammer K, Singer Y. Pranking with ranking//Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic. Vancouver, Canada, 2001: 641-647

[3] Herbrich R, Graepel T, Obermayer K, et al. Support vector learning for ordinal regression//Proceedings of the International Conference on Artificial Neural Networks. Edinburgh, UK, 1999: 97-102

[4] Burges C J, Shaked T, Renshaw E, et al. Learning to rank using gradient descent//Proceedings of the 22nd International Conference on Machine Learning. Bonn, Germany, 2005: 89-96

[5] Freund Y, Iyer R D, Schapire R E, et al. An efficient boosting algorithm for combining preferences. Journal of Machine Learning Research, 2003, 4(1): 933-969

[6] Cao Z, Qin T, Liu T, et al. Learning to rank: From pairwise approach to listwise approach//Proceedings of the International Conference on Machine Learning. Corvalis, USA, 2007: 129-136

[7] Yue Y, Finley T, Radlinski F, et al. A support vector method for optimizing average precision//Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval. Amsterdam, The Netherlands, 2007: 271-278

[8] Lai H, Pan Y, Tang Y, et al. FSMRank: Feature selection algorithm for learning to rank. IEEE Transactions on Neural Networks, 2013, 24(6): 940-952

[9] Gao W, Cai P, Wong K, et al. Learning to rank only using training data from related domain//Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Geneva, Switzerland, 2010: 162-169

[10] Cai P, Gao W, Zhou A, et al. Query weighting for ranking model adaptation//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, USA, 2011: 112-122

[11] Cai P, Gao W, Wong K, et al. Weight-based boosting model for cross-domain relevance ranking adaptation//Proceedings of the 33rd European Conference on Advances in Information Retrieval. Dublin, Ireland, 2011: 562-567

[12] Cai P, Zhou A. A novel framework for ranking model adaptation//Proceedings of the 2010 7th Web Information Systems and Applications Conference. Washington, USA, 2010: 149-154

[13] Chen D, Xiong Y, Yan J, et al. Knowledge transfer for cross domain learning to rank. Information Retrieval, 2010, 13(3): 236-253

[14] Goswami P, Amini M, Gaussier E, et al. Transferring knowledge with source selection to learn IR functions on unlabeled collections//Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. San Francisco, USA, 2013: 2315-2320

[15] Li P, Sanderson M, Carman M J, et al. On the effectiveness of query weighting for adapting rank learners to new unlabelled collections//Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. Indianapolis, USA, 2016: 1413-1422

[16] Goodfellow I J, Pougetabadie J, Mirza M, et al. Generative adversarial nets//Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal, Canada, 2014: 2672-2680

计算机学报

[17] Zhang H, Xu T, Li H, et al. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks//Proceedings of the International Conference on Computer Vision(ICCV). Venice, Italy, 2017: 5908-5916

[18] Arjovsky M, Chintala S, Bottou L, et al. Wasserstein generative adversarial networks//Proceedings of the 34th International Conference on Machine Learning. Sydney, Australia, 2017: 214-223

[19] Mao X, Li Q, Xie H, et al. Least squares generative adversarial networks//Proceedings of the International Conference on Computer Vision(ICCV). Venice, Italy, 2017: 2813-2821

[20] Zhu J, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks//Proceedings of the International Conference on Computer Vision(ICCV). Venice, Italy, 2017: 2242-2251

[21] Qin T, Liu T, Xu J, et al. LETOR: A benchmark collection for research on learning to rank for information retrieval. Information Retrieval, 2010, 13(4): 346-374

[22] Qin T, Liu T Y. Introducing LETOR 4.0 Datasets. arXiv preprint arXiv:1306.2597

[23] Benaim S, Wolf L. One-sided unsupervised domain mapping//Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA, 2017: 752-762

[24] Mackiewicz A, Ratajczak W. Principal components analysis (PCA). Computers and Geosciences, 1993, 19(3): 303-342

[25] Robertson S E. The probability ranking principle in IR. Journal of Documentation, 1977, 33(4): 294-304

[26] Järvelin K. IR evaluation methods for retrieving highly relevant documents//Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Athens, Greece, 2000: 41-48

ZHAO Wei-Qiang, M. S. candidate. His research interests include information retrieval, deep learning, etc.

ZHANG Xi, M. S. candidate. His research interests include multimodal retrieval, deep learning, etc.

LAI Han-Jiang, Ph.D., associate professor. His research interests include information retrieval, big data analysis, etc.

YIN Jian, Ph. D., professor. His research interests include data mining, big data analysis, etc.

《计算机学报》

Background

This work is supported by the National Key R&D Program of China (2018YFB1004404), and the National Natural Science Foundation of China (U1611264, U1711262, U1401256, U1501252, U1711261).

Cross-domain learning to rank is an important research topic in the field of information retrieval, which uses source domain data with labels to solve a ranking problem in target domain without labels. It can be applied in a wide range of applications, such as web search engines, recommendation systems, and product rating. However, most existing ranking adaptation frameworks are proposed based on the weighting scheme, which does not directly solve the problem that the data distributions in the source domain and the target domain are not the same. To this end, this paper proposes a cycle adaptation learning to rank, which is based on the generative adversarial networks and directly maps the source domain data to the target domain data. The feature distribution learning module is used to guide the learning of cross ranking module, and the cross ranking model can also give feedback for learning better-generated features.

Guangdong Key Laboratory of Big Data Analysis and Processing has three main research topics: big data storage and management, big data analysis and mining, and the advanced application of big data. Our lab has studied learning to rank since 2012. We have studied the sparse learning to rank algorithms, which are published in IEEE TNNLS, IEEE TC, and KBS. We also have studied the rank aggregation problem and the related finding is published in AAAI and CVPR.