

# Personalized Recommendation Model Based on Mini-Batch Gradient Descent

Xiaoran Bi, Shaoshan Yan, Ying Gao

Capital University of Economics and Business, Beijing  
Email: 18811383141@163.com

Received: Mar. 25<sup>th</sup>, 2019; accepted: Apr. 4<sup>th</sup>, 2019; published: Apr. 11<sup>th</sup>, 2019

---

## Abstract

Latent Factor Model (LFM) is one of the most widely used models in the recommendation system. The core idea is to connect users' interests and items through implicit features, obtain the objective function of users' interests relationship for items, and then obtain the optimal solution by Stochastic Gradient Descent, so as to make personalized recommendation for users. However, in the process of calculation, the stochastic gradient descent method will cause the value of the objective function to oscillate violently and the accuracy is insufficient, so we choose Mini-Batch Gradient Descent method to optimize the objective function, establish a model of personalized recommendation based on Mini-Batch Gradient Descent, reduce the randomness of the optimal solution of the objective function, and improve the accuracy and reduce the running time, so as to improve the quality of personalized recommendation. The experimental data adopted Movielens data set, Python as the tool, RMSE (root mean square error) and MAE (mean absolute error) as the standard, and compared the experimental results of several algorithms, verifying that the personalized recommendation model based on Mini-Batch Gradient Descent can get better recommendation effect.

## Keywords

Recommendation System, Latent Factor Model, Mini-Batch Gradient Descent, RMSE, MAE

---

## 基于小批量梯度下降法的个性化推荐模型

毕小然, 闫绍山, 高 迎

首都经济贸易大学, 北京  
Email: 18811383141@163.com

收稿日期: 2019年3月25日; 录用日期: 2019年4月4日; 发布日期: 2019年4月11日

## 摘要

隐语义模型(LFM)是推荐系统中应用比较广泛的模型之一。其核心思想是通过隐含特征联系用户兴趣和物品,得到用户对物品偏好关系的目标函数,然后通过随机梯度下降法求得最优解,从而个性化地对用户进行物品的推荐。但是计算过程中随机梯度下降法会造成目标函数值震荡的比较剧烈,准确度欠缺,所以提出对目标函数优化选择用小批量梯度下降法,建立基于小批量梯度下降法的个性化推荐模型,减少目标函数最优解的随机性,提高准确度,减少运行时间,从而达到提高个性化推荐质量的目的。实验数据采用Movielens数据集,Python作为工具,均方根误差(RMSE)、平均绝对误差(MAE)作为标准,将改进前后的算法结果做对比,验证基于小批量梯度下降法的个性化推荐模型能够得到更好的推荐效果。

## 关键词

推荐系统, 隐语义模型, 小批量梯度下降法, 均方根误差, 平均绝对误差

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

在个性化推荐中,最传统的推荐算法之一是协同过滤算法,通过用户对物品评分的评分矩阵形式进行运算。但是随着数据的增加,一个用户未评价的物品要远远多于评价的物品,因此就出现了数据稀疏问题。为了让算法能有一个更精准更高效的结果,在协同过滤算法中引入奇异值分解算法 SVD (Singular Value Decomposition),其推荐精度相对比传统的协同过滤算法有了一定的提高。在 Netflix 大奖赛中,Simon Funk [1]对 SVD 方法进行了改进,被称为 FunkSVD。主要过程是在 SVD 的目标函数中,使用随机梯度下降法使目标函数达到最优,后被称作是隐语义模型 LFM (Latent Factor Model)。

隐语义模型在个性化推荐领域中应用非常广泛。顾威[2]在音乐推荐系统的设计中使用了基于隐语义模型的推荐算法。张振领[3]在基于情境感知的个性化推荐研究中,以隐语义模型为基础,结合情境信息、用户行为、时间因子来预测用户对物品的真实兴趣度。古振威[4]在对人力资源推荐系统的研究中,提出了将隐语义模型与深度森林结合的算法来为用户提供更优质的人力资源推荐服务。肖迎元等人[5]在研究社交网络好友推荐方法中,采用隐语义模型来挖掘用户潜在兴趣特征和用户交友潜在偏好特征。

在隐语义模型纵向深入发展中,荆羽纯等人[6]提出了一种全新的基于学习自动机的矩阵训练算法。利用连续型学习自动机代替原有的梯度下降算法对大型稀疏矩阵的奇异值进行分解计算,提高了后续预测算法的精确度。但是基于学习自动机的推荐算法是在样本数较少的环境中,相比原算法有更精确的预测。王尔昕[7]设计了基于隐语义模型和聚类算法的优化算法,对传统算法中存在的稀疏性、时效性和冷启动等问题进行了改善。但使用聚类算法对用户进行聚类时,会将用户划分到固定的一类中来降低数据稀疏性,但是对于一些处于类别边界的用户,将他们划分的类别不一定能代表他们的特征,这导致在一些聚类数目下算法的准确度会降低甚至不如传统算法。鲁权等人[8]

提出融合隐语义模型与邻域模型的推荐算法, 将预测的精确度提高, 该方法针对隐式数据有明显的效果。王建芳等人[9]提出将奇异值分解技术与带偏置概率矩阵分解相结合的推荐方法, 缓解由数据高维稀疏性带来的推荐质量不高的问题。王科强[10]提出基于加权局部矩阵分解的商品推荐, 该模型设计了高效的子矩阵选择算法和改进的交替最小二乘参数优化算法, 对用户和商品的局部特性建模, 同时缓解了数据稀疏性问题。

在对目标函数求解的过程中。孙勇等人[11]在面向大规模服务性能预测的在线学习方法中, 提出了一种基于小批量在线学习的服务性能预测方法, 使大规模服务预测算法的时效性问题得到了有效的提升。Jing Li 等人[12]为解决大规模高光谱数据的分离问题中的重构误差最小化, 他们引入基于小批量梯度下降法来解决这个问题, 既降低了计算成本也对算法性能进行了优化。毛勇华等人[13]在用 BP 微调的深度学习贪婪层训练方法中, 将 BP 算法中三种梯度下降方法的实验结果进行了对比与分析, 实验结果表明在静态离线学习系统中, 采用随机小批量梯度下降可以得到最优解。

根据以上文献, 可以得知小批量梯度下降法对大规模数据的处理, 能够减少时间的运行, 优化推荐结果。所以本文提出在隐语义模型的基础上进行算法模型优化, 通过改进目标函数的优化过程来进一步提高算法准确度, 即将随机梯度下降法, 用同属于机器学习领域的小批量梯度下降法来代替。

## 2. 隐语义模型

隐语义模型 LFM [14]的基本思想是通过引入隐含特征, 将用户-物品的评分矩阵  $R$  降阶为两个矩阵的乘积,  $P$  表示用户对不同隐含特征的兴趣度,  $Q$  表示物品与不同隐含特征之间的关联程度, 这样可以将原评分矩阵的空缺评分数据补全, 实现用户对各个商品不同兴趣度的预测。由于评分矩阵非常庞大, 通常采用随机梯度下降法, 通过多次迭代来优化降阶矩阵  $P$ 、 $Q$  的参数, 从而得到最优解。

用户  $U$  对物品  $I$  的评分矩阵  $R$  计算公式如下:

$$\hat{R}_{UI} = P_U Q_I = \sum_{k=1}^K P_{UK} Q_{KI} \quad (1)$$

$\hat{R}_{UI}$  是用户  $U$  对物品  $I$  的兴趣度,  $K$  代表包含的隐含特征的数量。  $P_U$  和  $Q_I$  是模型参数的特征向量, 参数  $P_U$  和  $Q_I$  可以使用损失函数来进行训练。

$$C = \sum_{(U,I) \in K} (R_{UI} - \hat{R}_{UI})^2 + \lambda \|P_U\|^2 + \lambda \|Q_I\|^2 \quad (2)$$

上式中的  $\sum_{(U,I) \in K} (R_{UI} - \hat{R}_{UI})^2$  代表真实评分减去预测评分的误差平方和, 越小越好。  $\lambda \|P_U\|^2 + \lambda \|Q_I\|^2$  是用来防止过拟合的正则化项,  $\lambda$  是正规化因子, 需要根据特定应用环境反复实验得到。

接下来通过随机梯度下降算法来优化损失函数, 首先求参数  $P_{UK}$  和  $Q_{KI}$  的偏导数来确定梯度下降方向:

$$\frac{\partial C}{\partial P_{UK}} = -2(R_{UI} - \hat{R}_{UI})Q_{KI} + 2\lambda P_{UK} \quad (3)$$

$$\frac{\partial C}{\partial Q_{KI}} = -2(R_{UI} - \hat{R}_{UI})P_{UK} + 2\lambda Q_{KI} \quad (4)$$

其次通过不断的迭代计算来优化参数, 直至参数收敛:

$$P_{UK} = P_{UK} + \alpha \left( (R_{UI} - \hat{R}_{UI})Q_{KI} - \lambda P_{UK} \right) \quad (5)$$

$$Q_{KI} = Q_{KI} + \alpha \left( (R_{UI} - \hat{R}_{UI})P_{UK} - \lambda Q_{KI} \right) \quad (6)$$

$\alpha$  是学习速率, 用来控制每次迭代的步长。

### 3. 加入偏置项的隐语义模型

在实际情况中, 用户之间打分的差异或者项目本身质量的差异, 会导致评分系统中分数差异较大。所以通过在隐语义模型中加入偏置项[15]来消除这一差异:

$$b_{ui} = \mu + b_u + b_i \quad (7)$$

其中,  $\mu$  是全局平均值,  $b_u$  是用户偏置项,  $b_i$  是项目偏置项。

预测评分被定义为:

$$\hat{R}_{UI} = \mu + b_u + b_i + P_U Q_I \quad (8)$$

目标函数为:

$$C = \sum_{(U,I) \in K} (R_{UI} - (\mu + b_u + b_i + P_U Q_I))^2 + \lambda \|P_U\|^2 + \lambda \|Q_I\|^2 + \lambda \|b_u\|^2 + \lambda \|b_i\|^2 \quad (9)$$

同理, 求出四个参数的偏导数, 然后更新参数为:

$$Q_{KI} = Q_{KI} + \alpha \left( (R_{UI} - \hat{R}_{UI}) P_{UK} - \lambda Q_{KI} \right) \quad (10)$$

$$P_{UK} = P_{UK} + \alpha \left( (R_{UI} - \hat{R}_{UI}) Q_{KI} - \lambda P_{UK} \right) \quad (11)$$

$$b_u = b_u + \alpha \left( (R_{UI} - \hat{R}_{UI}) - \lambda b_u \right) \quad (12)$$

$$b_i = b_i + \alpha \left( (R_{UI} - \hat{R}_{UI}) - \lambda b_i \right) \quad (13)$$

### 4. 小批量梯度下降法

下降梯度法分为随机下降梯度法、批量下降梯度法和小批量梯度下降法。

随机梯度下降法(Stochastic Gradient Descent)是对于每个数据都计算损失函数, 然后求梯度更新参数。此方法有较快的计算速度, 但是收敛性能较差, 可能会在最优点附近左右摇摆, 无法得到最优解。

批量下降梯度法(Batch Gradient Descent)是遍历所有数据后算一次损失函数, 然后求解梯度更新参数。此方法可以得到全局最优解, 但是每次更新参数时, 都要把所有数据看一遍, 导致了计算速度慢, 计算量大。

为了克服两种方法的缺点, 将两种办法进行了融合提出了小批量梯度下降法(Mini-Batch Gradient Decent), 这种方法在数据中每次随机抽取一小批数据, 这样通过一批数据共同决定梯度方向, 下降起来更加稳定, 降低了收敛波动性。同时也减轻计算量, 减少运行时间。

### 5. 基于小批量梯度下降法的个性化推荐模型

在增加偏置项隐语义模型的基础上, 将小批量梯度下降法融入其中。其主要步骤是: 将随机选择一定批量的数据, 通过这一批量数据来共同决定步长的梯度方向, 以减少了随机性与震荡性。

度量标准采用均方根误差 RMSE (root-mean-square error)以及平均绝对误差 MAE (Mean Absolute Deviation)。假设预测的用户评分矩阵为  $\hat{R}_{UI}$ , 对应的实际用户评分矩阵为  $R_{UI}$ ,  $N$  为用户进行打分物品的个数。RMSE 计算公式:

$$\text{RMSE} = \sqrt{\frac{\sum (R_{UI} - \hat{R}_{UI})^2}{N}}$$

MAE 计算公式:

$$\text{MAE} = \frac{\sum |R_{UI} - \hat{R}_{UI}|}{N}$$

两个度量值越小, 代表预测的精度越准确。

主要算法流程如下:

输入: 迭代步长 `steps`, 学习速率 `alpha`, 正规化因子 `Lambda`, 批量大小 `batch_size`

输出: RMSE, MAE

a) 定义模型 `model`:

定义全局偏置变量 `bias_global`

定义用户的偏好 `bias_user`

定义电影的偏好 `bias_item`

定义用户和电影一个批量的偏好 `batch_bias_user`, `batch_bias_item`

定义用户和电影的权重 `w_user`, `w_item`

给定批处理的用户和项的权重嵌入 `batch_w_user`, `batch_w_item`

b) 定义损失函数 `cost`:

数据损失(`data loss`)= `tf.nn.l2_loss(tf.subtract(真实值, 预测值))`

正则化损失(正则化项\*L2 惩罚方式)= `tf.multiply(regularizer, penalty)`

损失函数(`cost`) = 数据损失(`data loss`) + 正则化损失(正则化项\*L2 惩罚方式)

使用梯度下降训练 `train_loss = tf.train.GradientDescentOptimizer(学习速率 learning_rate= alpha).minimize(损失函数 cost)`

c) 设置两个迭代器生成随机批次, 用于训练和测试

d) 计算均方根误差 `RMSE = np.sqrt(np.mean(np.power((真实值-预测值), 2)))` 计算平均绝对误差 `MAE = abs(np.mean(真实值-预测值))`

## 6. 程序及结果

本次试验将采用 Movielens ml-1m 数据集(百万), 其包含用户、电影、以及用户对电影的评分。数据集详细信息如表 1 所示:

**Table 1.** Movielens ml-1m dataset overview

**表 1.** Movielens ml-1m 数据集概况

数据集	用户数目	电影数目	评分信息数目
Movies-100k	3900	6040	1,000,000

Spyder (Python3.6)作为工具, 来进行运行测试。将数据集以 8:2 的比例划分训练数据集和测试数据集。将隐语义模型, 增加偏置项隐语义模型以及基于小批量梯度下降法的个性化推荐模型三种方法运行结果进行对比。三种方法训练过程如图 1、图 2、图 3 所示, 横坐标为迭代步数(30 次), 纵坐标为 RMSE 和 MAE 的值。

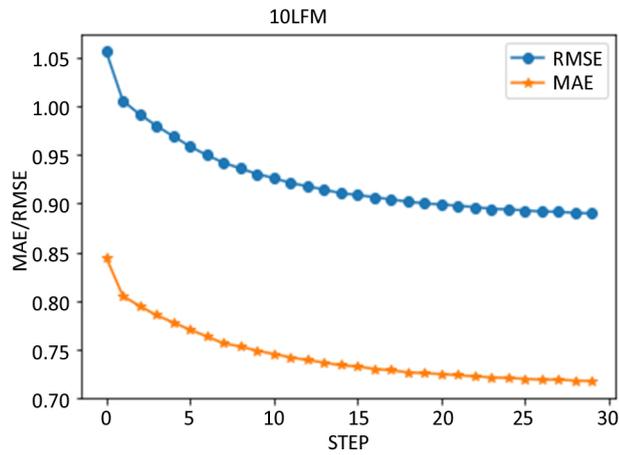


Figure 1. Latent factor mode  
图 1. 隐语义模型

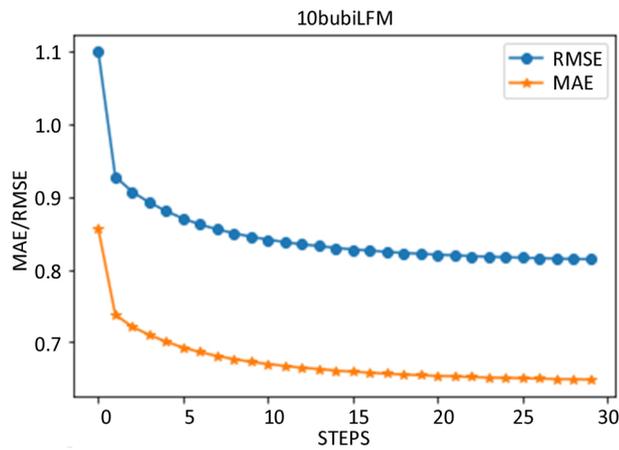


Figure 2. Latent factor model based on bias  
图 2. 增加偏置项隐语义模型

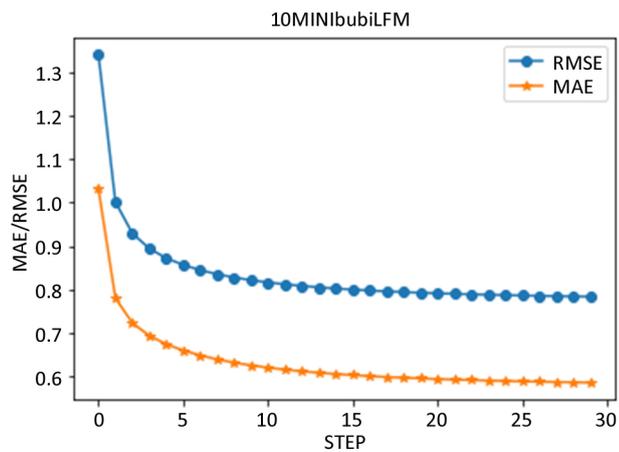


Figure 3. Personalized recommendation model based on Mini-Batch Gradient Descent  
图 3. 基于小批量梯度下降法的个性化推荐模型

三种模型实验结果如表 2 所示:

**Table 2.** Experimental results of the three models)**表 2.** 三种模型实验结果

10 万数据集	隐语义模型 LFM $A = 0.05$	增加偏置项隐语义模型 $A = 0.05$	基于小批量梯度下降法的个性化推荐模型 $A = 0.0018$
RMSE	0.956	0.920	0.903
MAE	0.782	0.728	0.714
Time	519.53 secs	569.47 secs	20.41 secs

通过图片可以看出三种方法其训练过程都逐渐趋于收敛。在实验结果中，可以看出每种方法求得最优解的  $\alpha$  参数都不同，这是一个需要根据实验环境不断调整的重要参数，如果  $\alpha$  参数设置不适当，很有可能会导致出现 Nan 值等问题。对比三种方法的 RMSE、MAE 值，基于小批量梯度下降法个性化推荐模型要低于前两者，说明此模型的精确度是优于前两者。同时三种方法在运行时间上有着鲜明的对比，在百万数据集的运行中，传统的隐语义模型需要约 519 秒的时间，增加偏置项隐语义模型由于增加了两个参数，虽然精度提高了，但是运行时间也增加了，需要约 569 秒的时间。而基于小批量梯度下降法个性化推荐模型的运行时间大幅度减少，只需要约 20 秒，平均每次循环只需要 0.6 秒，大大提高了运行效率，节省了时间。

基于小批量梯度下降法的推荐预测模型涉及的另一个重要参数便是批量的大小 batch size，设置合理的批量参数可以提高算法的性能效率，通常会设置 2 的幂数[16]来迎合 cpu gpu 的内存要求，从而获得最佳的训练稳定性和较高的运行速率。设置不同大小的批量规模进行实验，对比结果见表 3：

**Table 3.** Experimental results of operation in different batch sizes)**表 3.** 不同批量规模运行实验结果

基于小批量梯度下降法的预测模型	batch size = 64 ( $\alpha = 0.0015$ )	batch size = 128 ( $\alpha = 0.0013$ )	batch size = 512 ( $\alpha = 0.0016$ )	batch size = 1024 ( $\alpha = 0.0018$ )	batch size = 2048 ( $\alpha = 0.0001$ )
RMSE	0.930	0.913	0.909	0.903	0.941
MAE	0.742	0.725	0.718	0.714	0.753
Time	182.55secs	90.83 secs	48.74 secs	20.41 secs	14.33 secs

由实验结果可以看出，当批量设置较小 batch size = 64 或者过大 batch size = 2048 时，精确度结果都不尽人意。当 batch size = 1024 时，得到 RMSE = 0.903，MAE = 0.714 的最优值。由此可知只有选择了合理批量规模，才会使 RMSE 和 MAE 值达到最低，得到最优推荐精度。而对于基于小批量梯度下降法的个性化推荐模型的批量设置，应结合数据的规模以及设备的计算能力，通过多次试验，来选择合理的样本批量。

在运行时间中可以看出，批量设置的越大，运行时间越短。所以验证了在大规模数据下，和随机梯度下降相比，使用小批量(mini-batch)梯度下降方法的计算效率更高，减少很多运行时间，可以帮助快速训练模型。

## 7. 结论

在隐语义模型中，针对随机梯度下降法会造成目标函数震荡、准确度欠缺等问题，选择用小批量梯度下降法对目标函数进行优化，提出基于小批量梯度下降法的个性化推荐模型，该模型通过批量数据共同决定步长的梯度方向，减少了目标函数最优解的随机性与震荡性，提高准确度，同时缩短了大量运行时间。但是，如何有效快速地找到合理的超参数批量大小以及学习速率仍值得继续研究。

## 基金项目

北京市社会科学基金项目“‘互联网+’环境下北京公共信息流动机制及协同获取模式研究”(No. 16SRB021)。

## 参考文献

- [1] Netflix Update: Try This at Home. <http://sifter.org/simon/journal/20061211.html>
- [2] 顾威. 基于 Spark 的音乐推荐系统的设计与实现[D]: [硕士学位论文]. 哈尔滨: 哈尔滨工业大学, 2018.
- [3] 张振领. 基于情境感知的个性化推荐研究[D]. 天津: 天津理工大学, 2018.
- [4] 古振威. 基于隐语义模型与深度森林的人力资源推荐算法[D]. 广州: 华南理工大学, 2018.
- [5] 肖迎元, 张红玉. 基于用户潜在特征的社交网络好友推荐方法[J]. 计算机科学, 2018, 45(3): 220-224+254.
- [6] 荆羽纯, 葛昊, 江文, 王伊凡. 一种基于学习自动机的推荐算法改进[J]. 计算机应用研究, 2016, 33(1): 32-34+41.
- [7] Wang, E., Yao, W. and Wang, D. (2017) Collaborative Filtering Recommendation Algorithm Optimization Based on Latent Factor Model Clustering. *13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, Guilin, 29-31 July 2017, 1715-1719.
- [8] 鲁权, 王如龙, 张锦, 丁怡. 融合邻域模型与隐语义模型的推荐算法[J]. 计算机工程与应用, 2013, 49(19): 100-103+134.
- [9] 王建芳, 张朋飞, 刘永利. 基于改进带偏置概率矩阵分解算法的研究[J]. 计算机应用研究, 2017, 34(5): 1397-1400+1414.
- [10] 王科强. 基于矩阵分解的个性化推荐系统[D]: [博士学位论文]. 上海: 华东师范大学, 2017.
- [11] 孙勇, 谭文安, 谢娜, 蒋文明. 面向大规模服务性能预测的在线学习方法[J]. 计算机科学与探索, 2017, 11(12): 1922-1930.
- [12] Li, J., Li, X. and Zhao, L. (2017) Unmixing of Large-Scale Hyperspectral Data Based on Projected Mini-Batch Gradient Descent. *International Journal of Wavelets, Multiresolution and Information Processing*, **15**, Article ID: 1750059. <https://doi.org/10.1142/S021969131750059X>
- [13] 毛勇华, 桂小林, 李前, 贺兴时. 深度学习应用技术研究[J]. 计算机应用研究, 2016, 33(11): 3201-3205.
- [14] Yang, K., Liu, S., Liu, T. and Wang, X. (2016) Automatic Clustering Algorithm for Movie Recommendation Based on LFM Model. *International Conference on Electronic Information Technology and Intellectualization*, Guangzhou, 18-19 June 2016, 657-663.
- [15] Koren, Y., Bell, R. and Volinsky, C. (2009) Matrix Factorization Techniques for Recommender Systems. *Computer*, **42**, 30-37. <https://doi.org/10.1109/MC.2009.263>
- [16] Masters, D. and Luschi, C. (2018) Revisiting Small Batch Training for Deep Neural Networks.

### 知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>  
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2161-8801, 即可查询
2. 打开知网首页 <http://cnki.net/>  
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: [csa@hanspub.org](mailto:csa@hanspub.org)