


(/index.htm)

Home (/) Competitions (/competition/gameList.htm) Learning Data Sets (/datalab/index.htm) Rewards Forum (/competitio

me (/)> (/index.htm) Forum (./detail\_v1.htm)> 天池医疗AI大赛[第一季] 宜远智能\_HKBU团队Rank7 解决方案 (含代码)

天池医疗AI大赛[第一季] 宜远智能\_HKBU团队Rank7 解决方案 (含代码)

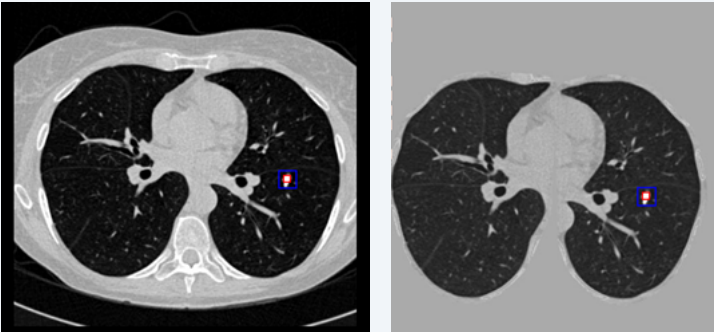
 JohnnyGambler Last comment 2017-10-09 14:43:48

主要参赛队员：  
吴宇，宜远智能co-founder  
施少怀，香港浸会大学博士（在读）， GPU高性能计算实验室  
陈晓辰，香港科技大学， 硕士

- .....
- 目录
- 1.数据预处理
  - 2.模型网络结构
  - 3.整体优化思路
    - 3.1 数据优化
    - 3.2 层次化Hard Mining
    - 3.3 LOSS 函数的设计
  - 4. 本次比赛的关键点总结
  - 5.关于模型收敛的思考
  - 6. 经验总结
- 代码开源说明

1.数据预处理

首先用SimpleITK把mhd图片读入，对每个切片使用Gaussian filter然后使用阈值-600把肺部图片二值化，然后再分析该切片的面和离心率大于0.99的区域，找到3D的连通区域。  
只保留0.68L到8.2L体积的区域，并且如果大于6000 mm2的区域到切片的中心区域的距离大于62mm也删除该连通区。最后只留7  
左边是原始图，右边是切完肺的。

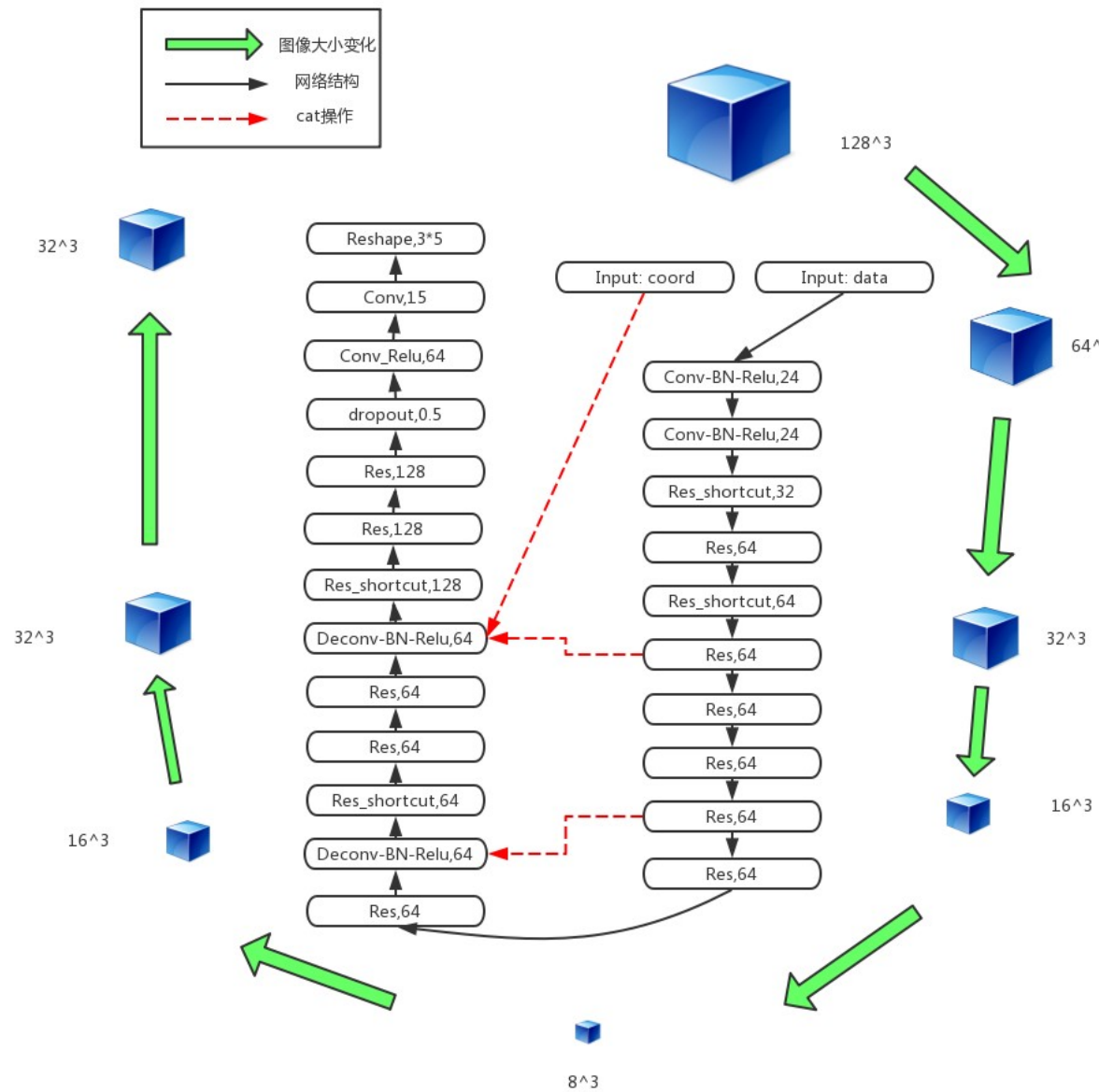


在实际中预处理中，我们可视化了每个肺的部分切片，存在一些bad case。主要有以下3种，我们也对这3种情况做了优化：  
1) 把肺边缘结节切掉。因为阈值导致的，把二值化环境-600改成-150有改善。  
2) 切出来全部为黑的(未找到任何肺部区域)。有些ct图是从头部开始扫描的，导致影响了连通区域判断，需要手动查看该mhd文件切片是肺部，在做完二值化操作后，人为把前面和后面的切片全部设置为0。  
3) 切出来只有一侧肺部情况。

有些患者两个肺的大小差别比较大，需要调整阈值，放宽阈值标注，把大于6000 mm2的区域到切片的中心区域的距离大于62mm mm2的区域到切片的中心区域的距离大于92mm也删除该连通区。并且在最后一步，不只保留最大的连通区，同时保留最大的两个连

2.模型网络结构

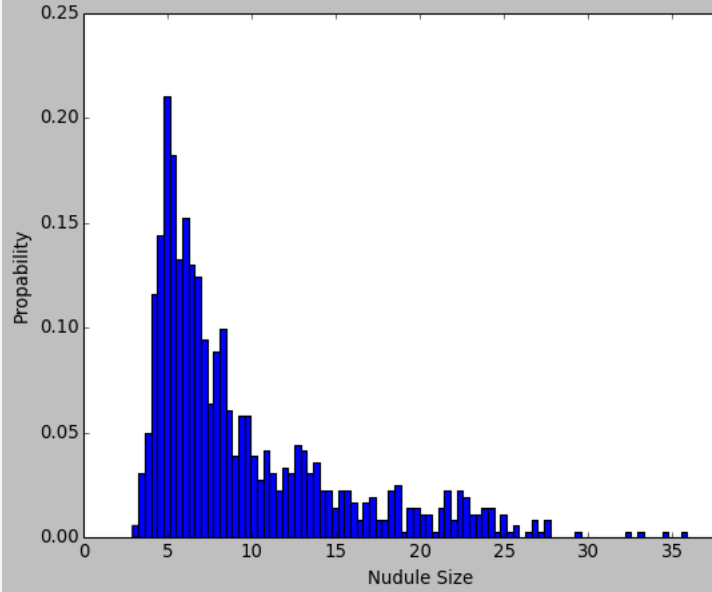
我们的网络如图所示，整体上是采用Unet+Resnet的思想。里面每个Resnet Block都是由多个卷积层和bn层和relu层组成的。我(整体深度大概150多层，这部分是参考了grt123的网络结构<https://github.com/lfz/DSB2017> (<https://github.com/lfz/DSB2017>))：



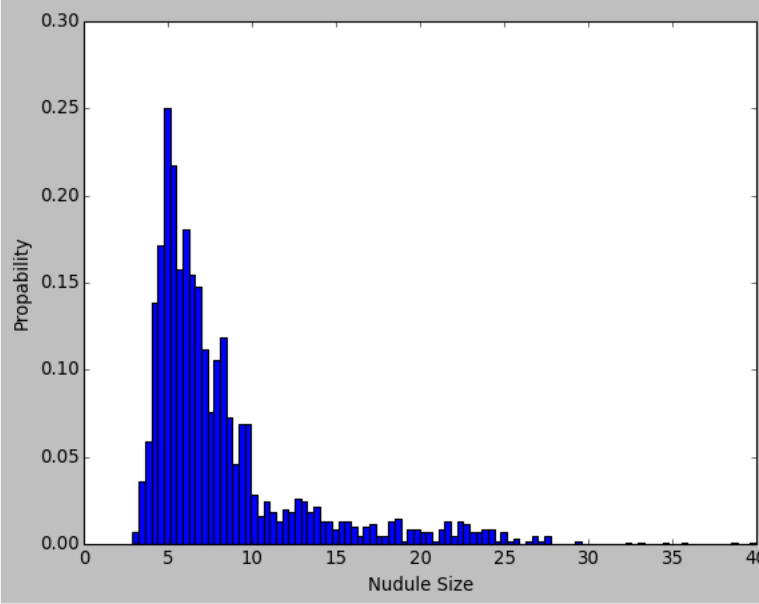
3.整体优化思路

3.1 数据优化

- 1) 肺部切割优化：这块其实没有完美的方法能把所有的肺一次性都切好。具体的思路我们已经在第 1 章数据预处理部分写出来了：切的不好的，再调参数重新切一次。
- 2) 10mm 以下结节的训练数据增强。我们在没做数据增强的情况下跑出来的模型，在验证集上漏掉了不少10mm以下的结节，所以整个框架在检测性能上得到很大的提升的主要贡献是来自于数据增强和Hard negative mining的调整。由于提供的数据并不多；验证集有200个病人共269个肺结节。结节的大小分布如图2所示。结节大小主要分布在4mm-30mm之间，以小结节为主。因此得比较准确才能把性能提升上去。肺结节大小分布如下：



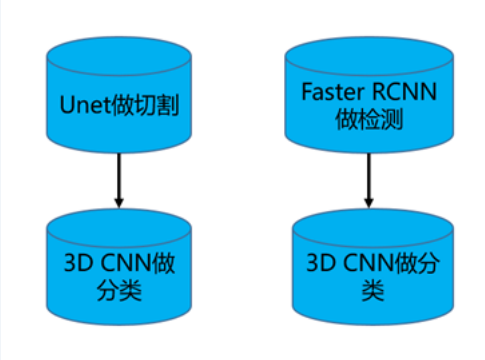
为了使网络模型对小结节更加敏感，我们对小结节进行数据增强，即增加小结节的数量，使得总体数据分布发生改变。以10mm为分段，对小于10mm的结节做4倍增强，即对于每1个小于10mm结节的正例，在训练时会用4个同样的结节做训练；对大于或等于10mm且小于30mm的结节做3倍增强；对大于或等于30mm且小于100mm的结节不做增强。增强后数据总量达到3280个结节。此处数据增强我们称为第一步增强



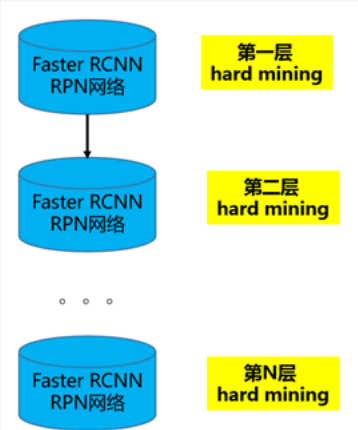
数据增强的目的是让神经网络去拟合小结节，对小结节更加敏感。在第一步增强的基础上进行模型训练，共进行100轮迭代，我们取80轮的模型（称为model80）进行FROC计算，在初赛验证集上平均和最高Score分别可达到0.65和0.70，测试集上Score为0.7

3.2 层次化Hard Mining

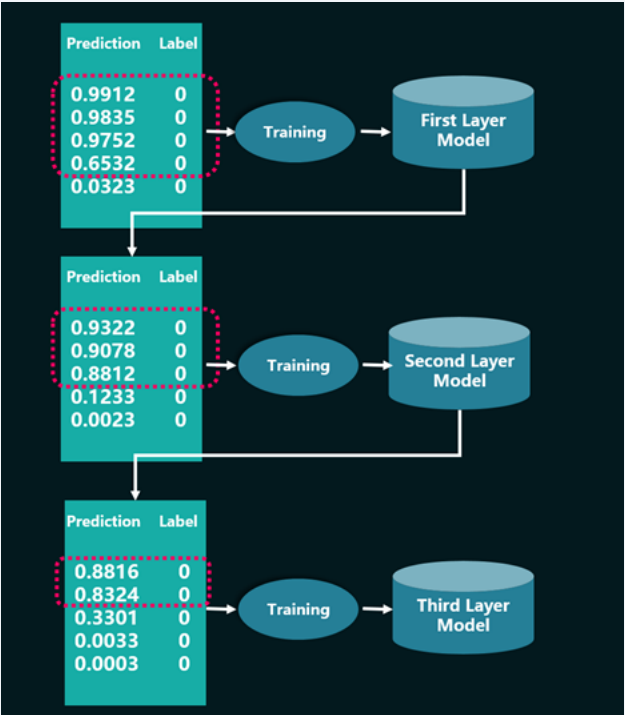
业界两套网络的做法比较普遍，比如用Unet切割或Faster RCNN检测，用3D CNN分类，如下图所示。



我们用的是如下统一的一套模型架构，即3D Faster RCNN的RPN网络，没有后续的全连接做分类，也并没有 再在后面接一套3D（节的网络参数。



该hard mining的过程，其实就是用上一层的模型作为下一层的输入，每一层的训练数据都选取比上一层更难分的。



这套架构，无需2套网络，只需要选择一套较深的网络。  
根据我们的经验，采取层次化模型训练。

第一赛季的Froc效果提升:

第一层: 0.733

第二层: 0.786

第三层: 0.806

第二赛季的Froc效果提升:

第一层: 0.621

第二层: 0.645 (时间来不及, 未完全收敛)

### 3.3 LOSS 函数的设计

在计算loss函数的时候, 我们做了2点优化。

1) 在使用hard mining的时候, 每个batchsize里面负例的个数会明显多于正例。为了防止算loss的时候被负例主导。我们将loss, 正例的loss和边框的loss。

$$L(\{p_i\}, \{t_i\}) = 0.5 * \frac{1}{N_{Pos}} \sum_i L_{pos\_cls}(p_i, p_i^*) + 0.5 * \frac{1}{N_{Neg}} \sum_i L_{Neg\_cls}(p_i, p_i^*) + \frac{1}{N_{reg}} \sum_i$$

2) 在上一节提到的层次化hard mining, 我们在最后一层训练模型的时候, 会修改loss函数的计算, 对于分错的负例和正例, 做像的。

$$L(\{p_i\}, \{t_i\}) = 0.5 * \frac{1}{N_{Pos}} \sum_i w_{pos_i} * L_{pos\_cls}(p_i, p_i^*) + 0.5 * \frac{1}{N_{Neg}} \sum_i w_{neg_i} * L_{Neg\_cls}(p_i, p_i^*) + \frac{1}{N_{reg}} \sum_i$$

比如:

红框里面的部分, 本来是负例, 却以很大的概率被分成正例, 这部分在算loss的时候权值就大些。红框外面的部分权值就小些。

Prediction	Label
0.9322	0
0.9078	0
0.8812	0
0.1233	0
0.0023	0

## 4. 本次比赛的关键点总结

1) 解决了基于Intel extended Caffe的150多层的 3D Faster RCNN RPN网络收敛问题。

可以从2个方向来解决(线下Phi卡平台均已验证过)。

a) 将 drop out设置为0.1。缺点是容易过拟合。

b) 先训练一个crop size为32的模型

用这个模型做pre train model, 训练crop size 64的模型

依次类推。

直到完成crop size为128的模型训练

由于时间关系, 我们并未比较这2种思路的效果。比赛中使用的是第1个思路, 收敛的更快些。

2) 提出层次化Hard Mining的训练框架。并没有采用常见的, unet做分割 + 3D CNN降假阳 或者 2d faster rcnn做检测 + 3D CI套网络。减少了需要调节的网络参数。

3) 重新设计了loss函数, 防止负例主导loss的计算, 并且在降低loss的过程中, 更聚焦于分错的训练样本。

## 5.关于模型收敛的思考

注意到最后一层的卷积层输出为bsx32x32x32x3x5, 其中bs为batch size, 一个sample里有3个anchors, 每个anchor最后缩放预测出5个值: (x', y', z', s', p'), 其中(x', y', z')为结节坐标偏移量, s'为结节大小, p'为结节的概率。对于每一个样本的Gr p=1为正例, p=-1为负例。由于每个病人切出来的128x128x128的数据块里通常有结节的个数都比较少, 即p=1的个数会相比p=-1的时候用到Negative hard mining技术, 对于p=-1的用例并不全部用来算loss, Negative hard mining的个数设为2或3, 即对于这几个选出来的负例是p'值最大的几个, 其他的样本就全部舍弃。这样做的主要目的是不会再负例Dominate了你的模型。但这样带来播的时候只有几个用例的梯度(其他舍弃的都为0), 会使得收敛速度变慢。

在我们初赛的Pytorch平台上, 训练集有600个病人共975个肺结节, 训练集有600个病人共975个肺结节, 为了使网络模型对小结数据增广, 即增加小结节的数据, 使得总体数据分布发生改变。以10mm, 30mm和100mm大小将结节进行分段, 对小于10mm的结10mm结节的正例, 在训练时会用4个同样的结节做训练; 对大于或等于10mm且小于30mm的结节做3倍增广; 对大于或等于30mm

增广后数据总量达到3280个结节，其中以0.3的概率随机切割负例，即总样本数为 $3280/(1-0.3)=4685$ 。Batch size为16，即平均每本和5个负样本，在进行Dense pixel label mapping时，会产生大量p=-1样本。假设Negative harding mining为2，每个batch贡献，为计算满 $32 \times 32 \times 32 \times 3 = 98304$ 个prediction，需要 $98304/33=2978$ 个batches，共约 $2978/(4685/16)=10$ 个epochs使模型开始收敛。但是在Intel Extended Caffe的平台上，和pytorch同样参数的条件下（不修改dropout为0.1）。基于3D Faster RCNN的150层96时，我们始终无法收敛。这是我们一直没想通的地方？

## 6. 经验总结:

我们团队虽然过往深度学习架构经验多，但对医学影像处理的know how属于尚在探索之中。所以，我们的优化思路，是用更多的构，来弥补没有专用模型网络积累的短板。在第一轮比赛时通过调用比较充足的计算资源时效果比较显著，但在第二轮限定计算资源的算资源及时间。

在计算资源比较充沛的情况下，选取比较深的Resnet效果会明显。在资源受限的实际场合或者现实的生产环境，我们有两点启发：

a) 学会认同重复造轮子的基础性工作。第一轮比赛我们是pytorch框架，第二轮按要求在caffe上实现，特别是在Intel Extend不少很基础的模块，这种貌似重复造轮子的工作，对我们提出了更高的要求，但也锻炼了我们深入到框架底层的能力，从而对不同框架重写甚至还因此帮我们找到我们第一版pytorch代码里detect部分存在的一个bug。

b) 根据资源灵活优化训练策略乃至模型。我们的3D Faster RCNN 初期在Extended Caffe 上过于耗时，但因为在计算资源充足以没有去考虑一些更快的检测算法，比如SSD、YOLO等，这也就是路径依赖的教训了。

### 代码开源说明:

由于时间关系，我们后面提交模型的时候都没有等到正常收敛。所以线下如果能完整的运行完层次化hard mining，效果应该要比我们目前在GitHub ([https://github.com/YiYuanIntelligent/3DFasterRCNN\\_LungNoduleDetector](https://github.com/YiYuanIntelligent/3DFasterRCNN_LungNoduleDetector))

([https://github.com/YiYuanIntelligent/3DFasterRCNN\\_LungNoduleDetector](https://github.com/YiYuanIntelligent/3DFasterRCNN_LungNoduleDetector)) 开源了核心代码，特别是将我们基于Intel Extended Caffe模块发布到社区，相信这也是业内首个Intel extended Caffe版的150层网络3D Faster RCNN开源，希望对Intel 的深度学习社区用户。该代码对医学影像的处理也展示了有效性，相信对医学影像领域AI实践的发展，对技术如何造福大众，能起到一些帮助。

通过开源，希望有同行提出性能优化、功能扩充等的修改建议，互相促进。对我们的开源代码及相关医学影像处理有任何疑问、建

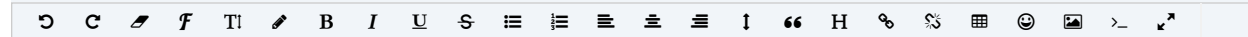
tkots\_wu@sina.com (mailto:tkots\_wu@sina.com) JohnnyGambler

cssshi@comp.hkbu.edu.hk (mailto:cssshi@comp.hkbu.edu.hk) 施少怀

[查看Gitlab代码](#)

## 复制创建PAI实验

☒ Rich Text



Topic Content...



liushrui:

宇哥给力

2017-10-12 09:39:47 1F 0  回复

```
(./new_edit
```



zlk411:

宇哥怎么标注啊

2017-10-16 12:17:52 2F 0  回复

```
(./new_edit
```

JohnnyGambler:回复@zlk411 (/science/scientistDetail.htm?userId=1095279142896) 具体是哪个标注

2017-10-16 12:46:50 3F 0 回复

[\(./new\\_edit\)](#)

zlk411:faster rcnn的标注

2017-10-16 17:40:15 4F 0 回复

[\(./new\\_edit\)](#)

JohnnyGambler:回复@zlk411 (/science/scientistDetail.htm?userId=1095279142896) 初赛的数据里面就有标注。你把坐标和结节大小提取出来据格式。

2017-10-16 18:03:28 5F 0 回复

[\(./new\\_edit\)](#)

zlk411:  
字哥，我现在只想把二维faster-rcnn做好。我现在手里有lids数据库，dicom文件。但是之前我都是将dicom转成png，然后用label-master在png能到74%了，测试200张。可是我现在想直接将dicom送入faster-rcnn，可不会标注dicom,生不成xml就没办法训练，想问问字哥是怎么做的，而

2017-10-17 21:09:31 6F 0 回复

[\(./new\\_edit\)](#)

JohnnyGambler:回复@zlk411 (/science/scientistDetail.htm?userId=1095279142896) 我们没有按faster rcnn的原始的模板生成 x m l 文件。自己节坐标位置和结节大小。另外，建议你转 3 D 的模型，我们线下的pytorch训练框架，做完第一层训练，准确率就能到97%。

2017-10-18 13:32:32 7F 0 回复

[\(./new\\_edit\)](#)

KeeganLi:666

2017-10-20 10:08:03 8F 0 回复

[\(./new\\_edit\)](#)

风的羽毛:  
请问赛题数据可以分享下么，多谢多谢~

2017-10-20 11:26:52 9F 0 回复

[\(./new\\_edit\)](#)

JohnnyGambler:回复@风的羽毛 (/science/scientistDetail.htm?userId=1095279172387) 我问下官方，看有没有版权问题。如果O K的话，我找的。BTW，你如果着急的话，可以去先下载luna的数据，格式和这个比赛貌似是一样的，而且也不用切肺。https://luna16.grand-challenge.org/

2017-10-20 12:09:20 10F 0 回复

[\(./new\\_edit\)](#)

<<

<

1

2

>

>>

[About us \(/aboutUs.htm\)](#)      [Legal agreement \(/userAgreement.htm\)](#)

Explore Alibaba Cloud	Service and Support	Pricing	Contact Us	User Center
Why Alibaba Cloud (//www.alibabacloud.com/why-alibaba-cloud)	Alibaba Cloud Support (//www.alibabacloud.com/support/sales)	Pricing Models (//www.alibabacloud.com/pricing)	Contact Sales (//www.alibabacloud.com/contact-sales)	Console (//home.console.aliyun.com)
Free Credit (//www.alibabacloud.com/campaign/free-trial)	Documentation (//www.alibabacloud.com/help)		Report Abuse (//www.alibabacloud.com/report)	Billing Management (//billing.console.aliyun.com)
Products (//www.alibabacloud.com/products)	Getting Started (//www.alibabacloud.com/getting-started)			Account Management (//account-intl.console.aliyun.com)

Customers (//www.alibabacloud.com/customers)	Forum (//www.alibabacloud.com/forum)	Ticket Management (//workorder-intl.console.aliyun.com)
Partners (//www.alibabacloud.com/partners)	ICP Support (//www.alibabacloud.com/icp)	Message Center (//notifications-intl.console.aliyun.com)
Startups (//www.alibabacloud.com/startups)	WHOIS (//www.alibabacloud.com/whois)	
Marketplace (//www.alibabacloud.com/marketplace)	Refer a Friend (//www.alibabacloud.com/campaign/referral)	
About Us (//www.alibabacloud.com/about)	Privacy Policy (https://www.alibabacloud.com/help/en/faq-detail/42425.htm)	
ChinaConnect (//www.alibabacloud.com/help/product/42384.htm)	Training & Certification (//www.alibabacloud.com/support/tutor-program)	Notice List (//www.alibabacloud.com/notice)
Alibaba Group (http://www.alibabagroup.com/en/global/home)	Taobao Marketplace (//www.taobao.com/)	Tmall (//www.tmall.com/)
Cloud Focus (http://www.aliexpress.com/)	Alibaba.com (http://www.alibaba.com/)	Juhuasuan (http://www.alimama.com/index.htm)
Alibaba.com (http://www.alibaba.com/)	1688 (http://www.1688.com/)	Alimama (http://www.alimama.com/index.htm)
AliTelecom (http://www.aliqin.cn/)	AutoNavi (http://www.autonavi.com/)	UCWeb (http://www.ucweb.com/)
Ur (http://www.xiami.com/)	TTPod (http://www.ttpod.com/)	DDChong (http://www.ddchong.com/)
DingTalk (http://www.dingtalk.com/?lwfrom=2017-2017)	UC Union (http://uc-union.com/)	Lazada (http://taobao.lazada.sg)
© 2009-2017 Copyright by Alibaba Cloud All rights reserved		
Sitemap (//www.alibabacloud.com/sitemap.html)		