

导航

- 博客园
- 首 页
- 新随笔
- 联 系
- 订 阅 XML
- 管 理

< 2018年9月 >						
日	一	二	三	四	五	六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

公告

昵称：袁军峰  
园龄：9年9个月  
粉丝：7  
关注：7  
+加关注

搜索

找找看

谷歌搜索

常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签

我的标签

- Linux(7)
- CUDA(3)
- MTK(2)
- python(2)
- SQL(2)
- 机器学习(2)
- sed(2)
- shell(1)
- 进程地址空间(1)
- 库使用(1)
- 更多

随笔分类

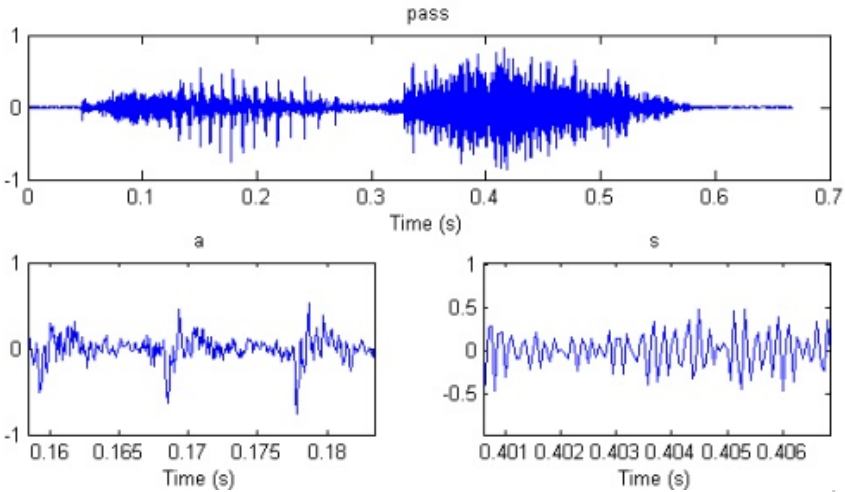
- android(2)
- ASP.net(2)
- C++学习(12)
- C语言学习(26)
- Flash学习(2)
- HTTP协议(1)
- iPhone开发(1)
- JAVA学习(3)
- Linux学习(61)
- MTK平台学习(6)
- python脚本(19)
- svn使用(8)
- 并行计算(10)

波形、频谱和语谱

1.声音最直接的表示方式是**波形**，英文叫**waveform**，就是你贴的左边那张图。另外两种表示方式（频谱和语谱图）下文再说。波形的横轴是时间（所以波形也叫声音的**时域**表示），纵轴的含义并不重要，可以理解成位移（声带或者耳机膜的位置）或者压强。

当横轴的分辨率不高的时候，语音的波形看起来就是像你贴的图中一样，呈现一个个的三角形。这些三角形的轮廓叫作波形的**包络（envelope）**。包络的大小代表了声音的响度。一般来说，每一个音节会对应着一个三角形，因为一般地每个音节含有一个元音，而元音比辅音听起来响亮。但例外也是有的，比如：1）像/s/这样的音，持续时间比较长，也会形成一个三角形；2）爆破音（尤其是送气爆破音，如/p/）可能会在瞬时聚集大量能量，在波形的包络上就体现为一个脉冲。

下面这张图中上方的子图，是我自己读单词pass /pæs/的录音。它的横坐标已经被我拉开了一些，但其实这个波形是由两个“三角形”组成的。0.05秒处那个小突起是爆破音/p/，0.05秒到0.3秒是元音/æ/，0.3到0.58秒是辅音/s/。



如果你把横轴的分辨率调高，比如只观察0.02s秒甚至更短时间内的波形，你就可以看到波形的**精细结构（fine structure）**，像上图的下面两个子图。波形的精细结构可能呈现两种情况：一种是有周期性的，比如左边那段波形（图中显示了周期多一点），这种波形一般是元音或者辅音中的鼻音、浊擦音以及/l/、/r/等；另一种是乱的，比如右边那段波形，这种波形一般是辅音中的清擦音。辅音中的爆破音，则往往表现为一小段静音加一个脉冲（如pass开头的/p/）。

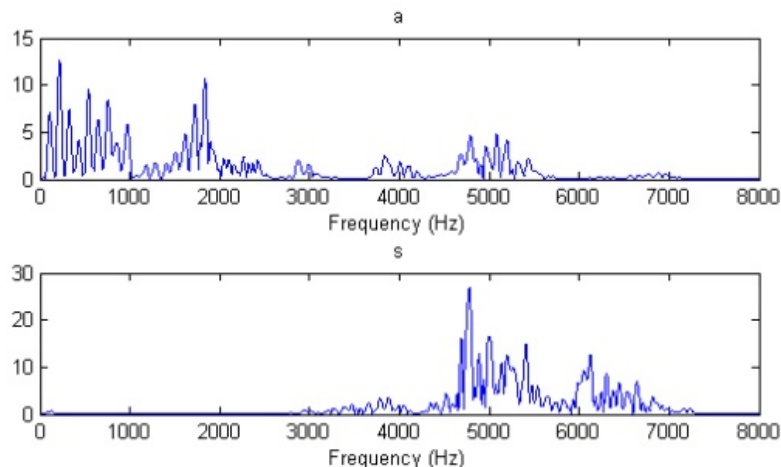
2. 看完了声音的**时域**表示，我们再来看它的**频域**表示——**频谱（spectrum）**。它是由一小段波形做傅里叶变换（Fourier transform）之后取模得到的。注意，必须是一小段波形，太长了弄出来的东西（比如你贴的右边的图）就没意义了！这样的一小段波形（通常在0.02~0.05s这样的数量级）称为**一帧（frame）**。下面是我读的pass的波形中，以0.17s和0.4s为中心截取0.04s波形经傅里叶变换得到的频谱。频谱的横轴是频率；我录音的采样率用的是16000 Hz，频谱的频率范围也是0 ~ 16000 Hz。但由于0 ~ 8000 Hz和8000 ~ 16000 Hz的频谱是对称的，所以一般只

程序人生(8)  
程序优化(15)  
大数据(2)  
机器学习(13)  
脚本(4)  
数据库(1)  
图像处理(3)  
语音识别(7)

## 随笔档案

2018年8月(2)  
2018年4月(3)  
2018年3月(1)  
2017年12月(1)  
2017年11月(1)  
2017年6月(2)  
2017年4月(3)  
2017年3月(2)  
2017年1月(1)  
2016年12月(3)  
2016年10月(1)  
2016年9月(1)  
2016年8月(2)  
2016年7月(3)  
2016年6月(2)  
2016年5月(1)  
2016年4月(3)  
2016年2月(1)  
2016年1月(1)  
2015年10月(3)  
2015年9月(6)  
2015年8月(3)  
2015年7月(10)  
2015年6月(9)  
2015年5月(5)  
2015年4月(17)  
2015年3月(7)  
2015年2月(3)  
2015年1月(3)  
2014年12月(4)  
2014年11月(4)  
2014年10月(1)  
2014年9月(3)  
2014年7月(9)  
2014年6月(3)  
2014年5月(5)  
2014年4月(3)  
2014年3月(11)  
2014年2月(1)  
2014年1月(2)  
2013年12月(2)  
2013年10月(2)  
2013年8月(3)  
2013年7月(1)  
2012年12月(2)  
2012年4月(1)  
2012年3月(1)  
2012年1月(2)  
2011年12月(6)  
2011年11月(2)  
2011年10月(2)  
2011年9月(2)  
2010年9月(1)  
2010年7月(2)  
2010年4月(4)  
2009年12月(1)  
2009年11月(1)  
2009年9月(1)

画0 ~ 8000 Hz的部分。



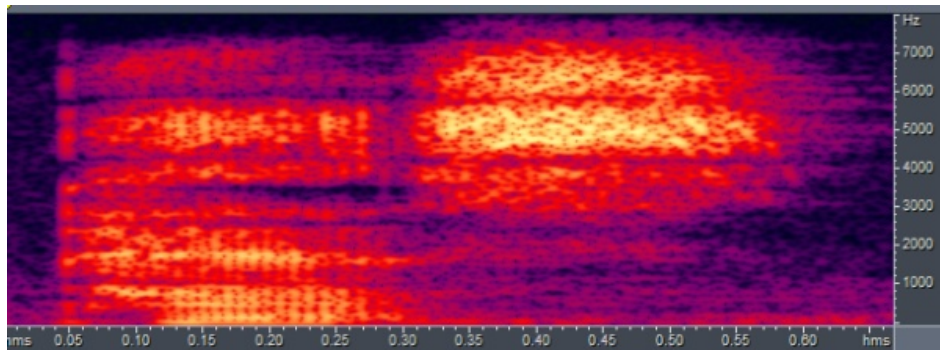
频谱跟波形一样，也有包络和精细结构。你把横轴压缩，看到的就是包络；把横轴拉开，看到的就是精细结构。我上面这两张图使得二者都能看到。

第一个频谱是元音/æ/的频谱，可以看到它的精细结构是有周期性的，每隔108 Hz出现一个峰。从这儿也可以看出来，语音不是一个单独的频率，而是由许多频率的简谐振动叠加而成的。第一个峰叫**基音**，其余的峰叫**泛音**。第一个峰的频率（也是相邻峰的间隔）叫作**基频（fundamental frequency）**，也叫**音高（pitch）**，常记作 $f_0$ 。有时说“一个音的频率”，就是特指基频。基频的倒数叫**基音周期**。你再看看上面元音/æ/的波形的周期，大约是0.009 s，跟基频108 Hz吻合。频谱上每个峰的高度是不一样的，这些峰的高度之比决定了**音色（timbre）**。不过对于语音来说，一般没有必要精确地描写每个峰的高度，而是用“**共振峰（formant）**”来描述音色。共振峰指的是包络的峰。在我这个图中，忽略精细结构，可以看到0~1000 Hz形成一个比较宽的峰，1800 Hz附近形成一个比较窄的峰。共振峰的频率一般用 $f_1$ 、 $f_2$ 等等来表示。上图中， $f_1$ 是多少很难精确地读出来，但 $f_2 \approx 1800\text{Hz}$ 。当然，在2800 Hz、3800 Hz、5000 Hz处还有第三、四、五共振峰，但它们与第一、二共振峰相比就弱了许多。除了元音以外，辅音中的鼻音、浊擦音以及/l/、/r/等也具有这种频谱，可以讨论基频和共振峰频率（不过浊擦音一般不讨论共振峰频率）。

第二个频谱是辅音/s/的频谱。可以看出它的精细结构是没有周期性的，所以就无所谓**基频**。一般也不提这种频谱的**共振峰**。清擦音的频谱一般都是这样。

### 2.5 在回答你的第三个问题之前，我们先来看一下声音的第三种表示方式——语谱图

**（spectrogram）**。上面说过，频谱只能表示一小段声音。那么，如果你想观察一整段语音信号的频域特性，要怎么办呢？我们可以把一整段语音信号截成许多帧，把它们各自的频谱“竖”起来（即用纵轴表示频率），用颜色的深浅来代替频谱强度，再把所有帧的频谱横向并排起来（即用横轴表示时间），就得到了语谱图，它可以称为声音的**时频域表示**。下面我就偷懒，不用Matlab自己画语谱图，而用Cool Edit绘制上面“pass”的语谱图，如下：



注意横轴是时间，纵轴是频率，颜色越亮代表强度越大。可以观察一下0.17s和0.4s处，是不是跟我上面画的频谱相似？然后再试着从这张语谱图上读出元音/æ/的第二共振峰频率。

语谱图的好处是可以直观地看出共振峰频率的变化。我上面读的“pass”中只有一个单元音，如果有双元音就会非常明显了。比如下面这张我读的“eye” /ai/，可以非常明显地看出在元音从/a/向/i/过渡的阶段（0.2 ~ 0.25s）， $f_1$ 在降低，而 $f_2$ 在升高。

2009年5月 (1)  
2008年12月 (1)

## 文章分类

JAVA学习(1)

## 相册

公式推导

## audio

### Deep Learning in NLP

这篇博客是我看了半年的论文后，自己对 Deep Learning 在 NLP 领域中应用的理解和总结，在此分享。其中必然有局限性，欢迎各种交流，随便拍。

LSTM模型理论总结(产生、发展和性能等)

RNNLM Toolkit

Introduction Neural network based language models are nowadays among the most successful techniques for statistical language modeling. They can be easily applied in wide range of tasks, including automatic speech recognition and machine translation, and provide significant improvements over classic backoff n-gram models. The 'rnnlm' toolkit can be used to train, evaluate and use such models. The goal of this toolkit is to speed up research progress in the language modeling field. First, by providing useful implementation that can demonstrate some of the principles. Second, for the empirical experiments when used in speech recognition and other applications. And finally third, by providing a strong state of the art baseline results, to which future research that aims to "beat state of the art techniques" should compare to.

rnnlm源码分析(一)

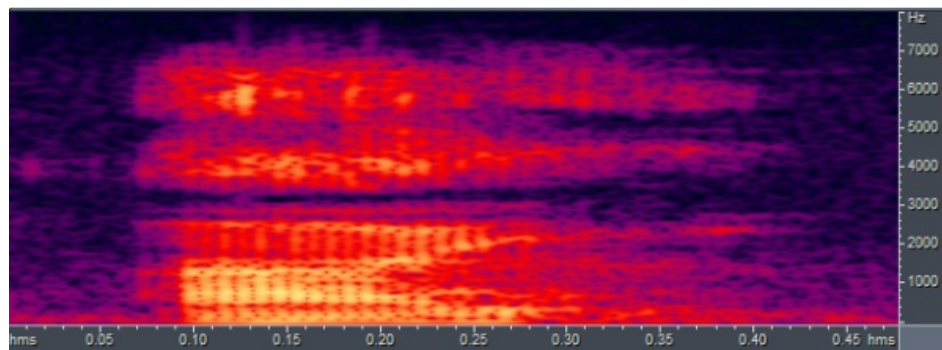
几个常见的语音交互平台的简介和比较

最近做了两个与语音识别相关的项目，两个项目的主要任务虽然都是语音识别，或者更确切的说关键字识别，但开发的平台不同，一个是 windows 下的，另一个是 Android 平台的，于是也就选用了不同的语音识别平台，前者选的是微软的 Speech API 开发的，后者则选用的是 CMU 的 pocketsphinx，本文主要将一些常见的语音交互平台进行简单的介绍和对比。

## BigData

大数据竞赛平台——Kaggle 入门篇

这篇文章适合那些刚接触 Kaggle、想尽快熟悉 Kaggle

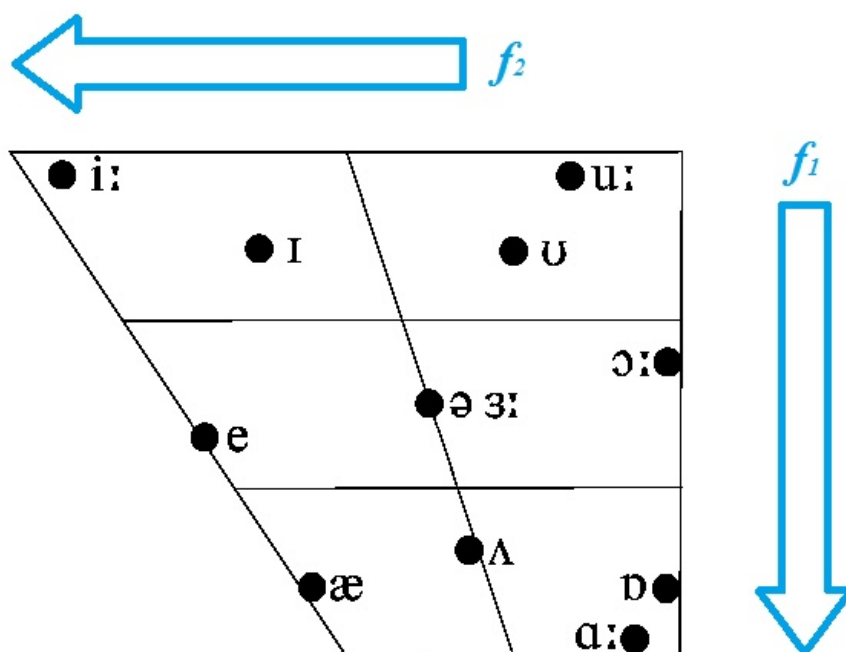


3. 元音与共振峰的关系已经研究得比较透彻了，简单地说：

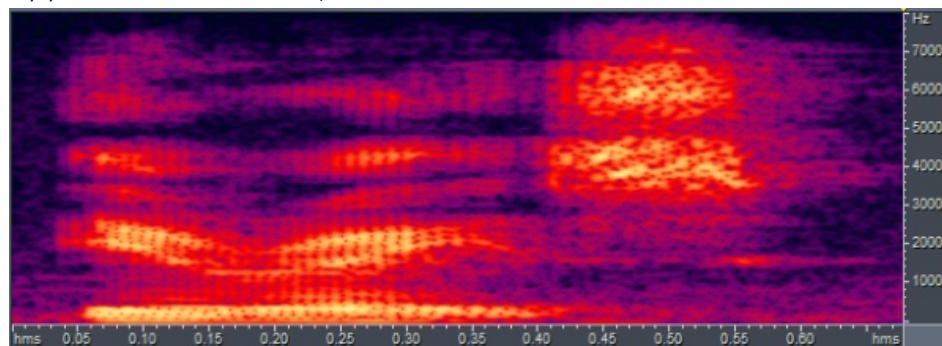
- 1) 开口度越大， $f_1$  越高；
- 2) 舌位越靠前， $f_2$  越高；
- 3) 不圆唇元音的  $f_3$  比圆唇元音高。

例如，/a/ 是开、后、不圆唇元音，所以  $f_1$  高， $f_2$  低， $f_3$  高；/y/（即汉语拼音的 ü）是闭、前、圆唇元音，所以  $f_1$  低， $f_2$  高， $f_3$  低。

也许题主见过下图那样的元音图（vowel chart），我把  $f_1$  和  $f_2$  的变化方向标了上去。



$f_3$  最明显的体现其实是在英语的辅音 /r/ 中，例如下面我读的 erase /i'reiz/ 的语谱图，可以看到辅音 /r/ 处（0.19s 左右） $f_3$  明显低，把  $f_2$  也压下去了。



清擦音可以根据能量集中的频段来分辨。下面是我读的 /f/, /θ/, /s/, /ʃ/ 的语谱图。浊擦音会在清擦



并且独立完成一个竞赛项目的网友，对于已经在Kaggle上参赛过的网友来说，大可不必耗费时间阅读本文。本文分为两部分介绍Kaggle，第一部分简单介绍Kaggle，第二部分将展示解决一个竞赛项目的全过程。如有错误，请指正！  
如何准备机器学习工程师的面试？

## linux编程

Linux静态库编译的问题  
解决libc.so.6: version 'GLIBC\_2.14' not found问题  
0.以下在系统CentOS 6.3 x86\_64上操作 1.试图运行程序，提示"libc.so.6: version 'GLIBC\_2.14' not found", 原因是系统的glibc版本太低，软件编译时使用了较高版本的glibc引起的：

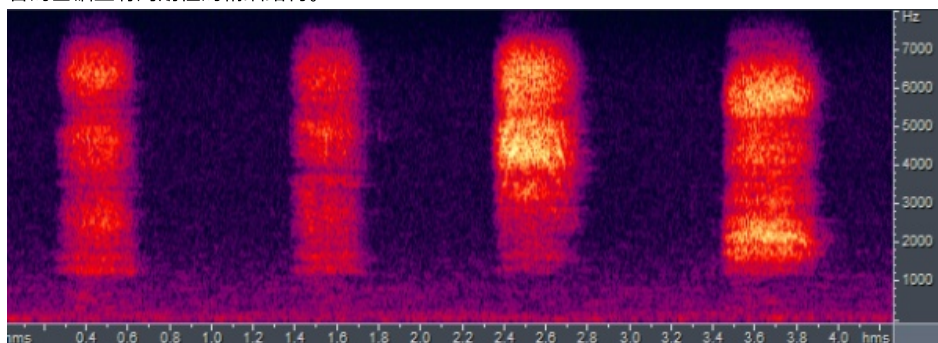
## open source code

github  
Git是Linux的作者Linus在2005年写的版本管理工具，它是一个分布式的工具，不区分客户端和服务端。代码库的每份拷贝都带有完整的数据库，用户可以在本地修改、提交代码，而代码库的不同拷贝之间，可以无缝地合并代码。DiBona很佩服GitHub的独到眼光：这就是Git的天才之处，而GitHub天才之处在于他们理解Git的价值。凭借Git，GitHub为所有的开源项目提供了一种类似于Linux内核的体验，人们可以随意克隆感兴趣的项目到自己的账户下，进行自己的修改，你可以长期维护自己的版本，定时和原作者的库进行同步，你也可以把自己的修改通过pull request的方式回馈给原作者。Git也为用户提供了私有仓库，这些仓库中的代码不会被公开。正如其名字所暗示的那样，GitHub正成为全世界开源软件的集中营，大家在这里以一种前所未有的高效的方式进行协作。几乎所有的公司都把它们开源项目放在了GitHub上，包括Google、Facebook、Twitter，甚至包括微软。微软最近开源了一系列他们最核心的软件，他们情愿使用GitHub，也不用自家的CodePlex服务。

## 机器学习

SVM参数设置  
LIBSVM使用方法及参数设置  
主要参考了一些博客以及自己使用经验。收集觉得比较有用的。  
WILDMML  
AI, DEEP LEARNING, NLP  
循环神经网络(RNN, Recurrent Neural Networks)介绍  
循环神经网络(Recurrent Neural Networks, RNNs)已

音的基础上有周期性的精细结构。



爆破音的爆破时间很短，在语谱图上一般较难分辨。

题主问的“两个音之间的音是什么样子”，就要分情况讨论了。

- 1) 如果是两个元音，那么可以在元音图上找到两个元音，取它们连线的中点。这对应着把 $f_1$ 、 $f_2$ 分别取平均。
- 2) 如果是两个清擦音，那么可以把它们的频谱取平均，这样听起来应该是个四不像（后来我做了实验，结果见这里：[Mixture of Unvoiced Fricatives](#)）。
- 3) 楼主提到的/t/和//属于不同类型的辅音，很难定义它们“之间”是什么东西。

链接：<https://www.zhihu.com/question/27126800/answer/35376174>

分类：[语音识别](#)

[好文要顶](#) [关注我](#) [收藏该文](#)  

 **袁军峰**  
关注 - 7  
粉丝 - 7

[+加关注](#)

[« 上一篇：可决系数R^2和MSE, MAE, SMSE](#)  
[» 下一篇：疑难错误之结果类型转换](#)

posted on 2017-04-21 09:31 袁军峰 阅读(797) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

- 【推荐】超50万VC++源码：大型组态工控、电力仿真CAD与GIS源码库！
- 【免费】要想入门学习Linux系统技术，你应该先选择一本适合自己的书籍
- 【前端】SpreadJS表格控件，可嵌入应用开发的在线Excel
- 【直播】如何快速接入微信支付功能

**腾讯云**

**1元搭建视频通话应用**

享受 10000分钟 时长，满足 **小程序 APP/H5/PC** 端接入场景

[立即抢购](#)



## 最新IT新闻：

- 把13亿中国人拉到一个微信群会发生什么？腾讯给出了回答
- Adobe与云营销软件开发商Marketo谈判 商讨收购事宜
- Twitter法务主管：在适当情形 Twitter会屏蔽特朗普

经在众多自然语言处理 (Natural Language Processing, NLP) 中取得了巨大成功以及广泛应用。但是, 目前网上与RNNs有关的学习资料很少, 因此该系列便是介绍RNNs的原理以及如何实现。主要分成以下几个部分对RNNs进行介绍: 1. RNNs的基本介绍以及一些常见的RNNs(本文内容); 2. 详细介绍RNNs中一些经常使用的训练算法, 如Back Propagation Through Time(BPTT)、Real-time Recurrent Learning(RTRL)、Extended Kalman Filter(EKF)等学习算法, 以及梯度消失问题 (vanishing gradient problem) 3. 详细介绍Long Short-Term Memory(LSTM, 长短时记忆网络); 4. 详细介绍Clockwork RNNs(CW-RNNs, 时钟频率驱动循环神经网络); 5. 基于Python和Theano对RNNs进行实现, 包括一些常见的RNNs模型。

最新隐马尔可夫模型HMM详解  
隐马尔可夫模型 (Hidden Markov Model, HMM) 最初由 L. E. Baum 和其它一些学者发表在一系列的统计学论文中, 随后在语言识别, 自然语言处理以及生物信息等领域体现了很大的价值。平时, 经常能接触到涉及 HMM 的相关文章, 一直没有仔细研究过, 都是蜻蜓点水, 因此, 想花一点时间梳理下, 加深理解, 在此特别感谢 52nlp 对 HMM 的详细介绍。

## 健康

杨志英谈肝血管瘤的认识

## 最新评论

1. Re: 将openCV中的IplImage格式的图片显示到Picture控件上  
方法一bmpinfo需要分配空间, 调色板为256,  
BITMAPINFO \*pbminfo = (BITMAPINFO\*)malloc(sizeof(BITMAPINFOHEADER)+256\*size).....  
--esue
2. Re: 什么是DC?  
学习了, 说的不错!  
--listen80
3. Re: 在VC++中, 出现\_BLOCK\_TYPE\_IS\_VALID(!>nBlockUse)的错误  
那解决方法呢?  
--飞鸽传书
4. re: 卖毕设系统  
I Cow  
--会长

## 阅读排行榜

1. Linux下rm -rf删除文件夹报错 Device or resource busy(10919)

- 杭州一科技企业硅谷买楼建创新中心
- 微软悄悄地撤下有争议的Edge弹出广告 称这是Insider测试的一部分
- » 更多新闻...



华为全联接大会 | 上海 | 2018.10.10-12  
「大会门票+云服务器」专属套餐0.35折起



## 最新知识库文章:

- 为什么说 Java 程序员必须掌握 Spring Boot ?
- 在学习中, 有一个比掌握知识更重要的能力
- 如何招到一个靠谱的程序员
- 一个故事看懂“区块链”
- 被踢出去的用户
- » 更多知识库文章...

## 历史上的今天:

2016-04-21 Nginx 下配置SSL证书的方法

2. 五种主要多核并行编程方法分析与比较(7024)
3. 出现Fatal IO error 11 (资源暂时不可用) on X server :0.0.的可能原因及解决方案(4314)
4. 先验概率与后验概率的区别（老迷惑了）(4131)
5. SQLite在多线程环境下的应用(4072)

#### 评论排行榜

---

1. 在VC++中,出现\_BLOCK\_TYPE\_IS\_VALID(I>nBlockUse)的错误(1)
2. 卖毕设系统(1)
3. 将openCV中的IplImage格式的图片显示到Picture控件上(1)
4. 什么是DC? (1)

#### 推荐排行榜

---

1. 先验概率与后验概率的区别（老迷惑了）(1)