计算机网络

http状态码

1XX 接受的请求正在处理

2XX 请求正确处理 200 OK

3XX 重定向 302 Moved Permanently

4XX 请求错误 400 Bad Request

5XX 服务器错误 502 Bad Gateway

OSI体系结构

应用层 表示层 会话层 传输层 网络层 数据链路层 物理层

应用层：HTTP SMTP FTP HTTPS(HTTP+SSL)

传输层：TCP UDP

网络层：IPV4 IPV6 ARP ICMP IGMP。0

数据链路层：PPP CSMA

TCP/IP

可靠传输和不可靠传输

谁应该负责数据传输的可靠性?如果让网络来实现,那么意味着数据每跨越一个节点就需要进行校验,这样的要求对于网络内的节点(路由器)太高太昂贵,大大降低了数据传送的速度,而且即便网络可靠,发送节点和接受节点也可能出现差错(死机,程序运行出错误).因此,可靠传输的任务应该交给终端节点来负责,比如说主机发现书错误时,可以要求对方重传,由于网络是尽最大可能去交付数据,所以重传出现错误的概率就非常小.

TCP是在不可靠的端到端网络协议上实现的可靠数据传输协议. 换句话说网络层是不能保证接收方拿到的数据一定是有序,无丢失.

使用差错检测技术（如 CRC ），接收方的数据链路层就可检测出帧在传输过程中是否产生误码。

无线网络的标准

802.11b 2.4GHZ 11Mb/s

802.11a 5GHZ 54Mb/s
不同的标准有不同的频段和速度

我的笔记本电脑就是802.11ac

无线网络技术

CDMA GPS

计算机网络各层的作用

计算机网络各层的设备

物理层：中继器， 信号放大器；集线器，多个端口的集线器，可以连接多台计算机。

数据链路层：交换机

网络层：路由器，提供网络层的协议转换，并在不同网络之间存储和转发分组。

传输层：传输网关

数据结构

二叉树的定义与应用

操作系统

常见的计算机操作系统

android linux windows Mac IOS

进程和线程的区别

根本区别：进程是操作系统资源分配的基本单位，而线程是处理器任务调度和执行的基本单位

资源开销：每个进程都有　独立的代码和数据空间（程序上下文），程序之间的切换会有较大的开销；线程可以看做轻量级的进程，同一类线程共享代码和数据空间，每个线程都有自己独立的运行栈和程序计数器（PC），线程之间切换的开销小。

包含关系：如果一个进程内有多个线程，则执行过程不是一条线的，而是多条线（线程）共同完成的；线程是进程的一部分，所以线程也被称为轻权进程或者轻量级进程。

内存分配：同一进程的线程共享本进程的地址空间和资源，而进程之间的地址空间和资源是相互独立的

数据库原理和应用

常见的数据库

传统的关系型数据库：MySQL SQL Server Oracle

非关系型数据库：Redis key-value型数据库

关系型数据库中的关系

数据表之间有一定的关系

即一对一关系，多对一关系，多对多关系

范式

第一范式：属性不可分

第二范式：

什么是数据库设计

数据库设计就是在特定的应用场景下， 建立起最优的数据库模式，包括数据库和应用系统，使之有效地存储数据，满足用户的需求。流程包括，需求分析，概念结构设计，逻辑结构设计，物理结构设计，数据库的实施(DDL)和数据库的运行维护(DCL DML).

编译原理

编译器结构

前端

与源语言相关，字符流——词法分析器——词法单元流——语法分析器——语法树——语义分析器——语法树——中间代码生成器

后端

中间代码--机器语言，和机器相关

NFA/DFA（词法分析）

NFA 不确定性有限状态自动机

DFA 确定性有限状态自动机

正则表达式底层就是NFA或DFA

NFA可以转化为DFA

可以理解为只接受特定规则模式的字符流的一个黑盒

语法分析器

上下无关文法

自顶向下的语法分析

自底向上的语法分析

LR语法分析技术

编程语言

C++

数组名，可以理解为一个const指针，指向数组空间的首地址

声明使得程序知道这个东西，可以声明多次，而定义只能有一次。

形参是函数定义时用于接受实参的变量，实参是函数实际调用时传入和变量或者数据。

引用即使别名，必须被初始化，并且初始化后不能再改变；引用不能为空，指针可以。

面向过程的static，当变量声明为static时，将存放在全局数据区

面向对象的static，该变量由类的对象共有。类中的静态成员变量

switch能够接受整型，布尔类型的变量

堆溢出：申请动态内存太大

栈溢出：函数递归层次太深

define：预编译阶段进行值或者字符串的替换，没有类型检查

const：在编译运行起作用，会进行类型检查

抽象 封装 继承 多态

new/delete是关键字 malloc/free是库函数

new delete底层调用的还是malloc

重载：在同一作用域中，两个函数名相同，但是参数列表不同（个数，类型），返回值没有要求。

include头文件<>与""的区别：双引号包含的头文件，查找头文件的路径时优先查找当前头文件目录；尖括号包含的头文件，查找头文件的路径时优先查找编译器设置的头文件路径。

C++友元函数 在类外定义的,在类内声明的,加上了friend关键字的函数,它可以访问类的private和protected成员.

JAVA和C++的区别

1. C++是编译型语言，编译器把C++的源文件.h .hpp .cpp编译为可执行的二进制文件（.exe .out），JAVA是半编译半解释型语言。先把.java文件编译为字节码，然后由JVM来运行字节码。
2. 因此，C++要远快于JAVA
3. JAVA是纯面向对象的语言，C++不是，C++可以结构化程序设计,面向过程.
4. C++支持运算符重载 JAVA不支持
5. C++ 容易内存泄漏,没有内存回收机制,而JAVA有

C++11

nullptr 关键字 NULL -> 0(int)

= default

explicit 用来修饰构造函数,避免隐式转换

auto

for-each

JAVA优点

跨平台 一次编写到处运行

自动管理内存

没有指针

软件危机

泛指计算机软件的开发和维护过程中所遇到的一系列严重的问题

大型软件开发周期长,人力物力耗费都很大,以及软件本身的复杂性,使得开发出来的软件质量低,需求不匹配,代码也难以维护

杂乱

Transformer

编码器解码器，然后是一个多头自注意力层（query key values查询，相似度,相加注意力,点积注意力），然后残差连接，layerNorm，这样一共N层，

机器学习中的数学

1. 微积分，多元函数，导数，偏导数，梯度，多元函数求极值，雅可比矩阵，Hessian矩阵
2. 向量，矩阵的各种运算，范数 L1 L2， 二次型，奇异值分解
3. 随机变量，分布函数，概率密度函数，条件概率，贝叶斯公式，常用概率分布，正态分布，二项分布，最大似然估计

常见UML模型

用例图：从用户角度出发描述系统功能，谁使用系统，以及他们使用系统可以做什么

类图：描述系统中的类，以及各个类之间的关系。常见的关系有泛化（继承），实现（类与接口），关联，依赖（一个类需要另外一个类的协助）

状态图：描述类的对象所有的可能状态

序列图（顺序图）：表示参与者以一定的顺序步骤与系统的对象交互的模型，重点在于消息序列，强调消息是如何在对象之间被发送和接收的。

协作图：显示对象间的动态合作关系，可以看作类图和顺序图的交集。如果强调时间与顺序，则使用顺序图。如果强调上下级关系，则选择协作图。

部署图：用来表示系统的物理部署

IDE

CLion 自动补全 代码着色 断点 逐行调试 插件多 自动纠错 cmakelist自行配置 一个项目可以有多个main函数 占内存

VS

小程序 项目

页面生命周期 OnLoad OnReady OnShow

页面栈，框架以栈的形式维护了当前的所有页面

页面和HTML CSS JS一样

HTML主要一个盒子模型，width height border margin padding

有常用的API 页面路由navigateTo redictTo Post Get

树洞后端

Springboot+Mybatis+MySql

MVC

机器人视觉

YOLOv3 快 作者使用C语言来实现

工程经历 区别自己平时做着玩，做不出来也没有关系，而做这个机器人视觉是要在赛场上实地解决问题的。首先是配置开发环境，首先是安装双系统，然后安装显卡驱动，cuda, cudnn, 下载yolov3源码，编译，安装ROS等，由于软件本身的复杂性吧，这其中会遇到很多问题，重在考验一个人解决问题的能力。学习一下pasval voc数据集，用labelimg框图，训练自定义数据集，训练测试。最后进行封装为一个API(python)。实际运行的时候，我们会从摄像机这个ROS节点拿到他广播的数据，然后转化为数组，丢入yolov3中。

贝叶斯公式

数据加密

对称加密 DES AES

非对称加密 RSA

散列算法 MD5

缓冲区

缓冲区是一块特殊的内存区域，用这个空间来暂存输入或输出的数据。

缓冲池由多个缓冲区组成

为什么要引入缓冲？高速设备和低速设备不匹配，这样势必会让高速设备等待。

缓存

缓存是一个很大的概念。

CPU中的Cache，中文名叫高速缓冲存储器

Buffer的核心作用是用来缓冲，缓和冲击。比如你每秒要写100次硬盘，对系统冲击很大，浪费了大量时间在忙着处理开始写和结束写这两件事嘛。用个buffer暂存起来，变成每10秒写一次硬盘，对系统的冲击就很小，写入效率高了，日子过得爽了。极大缓和了冲击。

Cache的核心作用是加快取用的速度。比如你一个很复杂的计算做完了，下次还要用结果，就把结果放手边一个好拿的地方存着，下次不用再算了。加快了数据取用的速度。

简单来说就是buffer偏重于写，而cache偏重于读。

敏捷开发

团队成员少，团队成员都有一定的开发经验，对于使用的编程语言有一定的了解，需求明确，业务人员和开发人员一起工作，重视沟通，快速开发出产品原型，不断进行迭代。

毕业设计

YOLOv1

单阶段目标检测算法 快 将分类问题转化为回归问题

YOLOv1将一个张图片划分为S×S个grid cell，然后每个grid cell 生成两个bounding box，但是只有其中的一个bbox能够预测物体，也就是说，每个grid cell只能预测一个物体。如果一个物体的中心点落在这个grid cell里面，那么就由这个grid cell 来进行预测。损失函数主要有坐标回归误差，物体置信度误差，以及分类误差。

NMS 非极大值抑制（预测阶段）scores_threshold iou_threshold

IOU 交并比

YOLOv2

YOLOv2引入了anchor，anchor就是在先验经验的基础上，固定了一定宽高比的bbox，加入了BN，使用新的网络架构darknet-19，加入了多尺度训练，加入了paththough层（类似于resnet的shortcut）

YOLOv3

YOLOv3主干网络采用darknet-53（引入了resnet），提取出不同尺度的特征图，分别来预测大物体，中型物体和小物体，做了多尺度的特征融合。

其他

SSD对于每个像素都会生成一个anchor

RCNN通过Selective Search方法，从一张图片中生成约2K个Region Proposals.然后放入神经网络中提取特征，最后放入到SVM分类器中分类得到结果。其中NN和SVM是单独训练的。所以叫做两阶段目标检测。

常见英文缩写

进程 Process

线程 thread

句柄 handle

游标 cursor

缓存 cache

缓冲 buffer

TTL: Time To Live 生存时间

RTT Round Trip Time

NAS 网络附属存储 Network Attached Storage

MTU Maximum Transmission Unit

SSD Solid State Drive

HDD Hard Disk Drive

CIDR Classless Inter-Domain Routing　无类域间路由

NLP CV DNN CNN RNN LSTM FNN(Feedforward Neural Network)

DS: data science

CVPR: 计算机视觉与模式识别大会 pattern recognition

ROI: Region of Interest 感兴趣区域

RMSE: Root Mean Square Error 均方根误差

MSE: Mean Square Error 均方误差

MLP: 最大似然估计

PCA: 主成分分析 Principal Component Analysis

SVM: 支持向量机

SGD: 随机梯度下降

BGD: 批量梯度下降

IOU: 交并比

FCN: 全卷积网络

GAN: 生成式对抗网络

EDA: Exploratory Data Analysis 探索性数据分析

SQL: Structed Query Language 结构化查询语言

DQL：Data Query Language，数据查询语言；

DDL：Data Definition Language，数据定义语言；

DML：Data Manipulation Language，数据操作语言；

RAM Random Access Memory

ROM Read Only Memory

USB 通用串行总线 Universal Serial Bus-

HTML 超文本标记语言

XML 可扩展标记语言

ASIC 专用芯片

迭代的方式 iterative

时间复杂度 time complexity

空间复杂度 space complexity

约束 constraint

compromise 折中

中庸 The Doctrine of Mean

线性代数 Linear algebra [ˈældʒɪbrə]

概率论 probability theory