

zzwu的专栏

目录视图 摘要视图 RSS 订阅

个人资料



访问：728581次
积分：8894
等级：BLOG 5
排名：第1124名

原创：137篇 转载：98篇
译文：119篇 评论：259条

文章搜索

文章分类

- IOI2006竞赛题 (9)
- LSI与PCB设计 (6)
- 人工智能和机器思维 (1)
- 图形图像算法 (7)
- 射影几何入门(连载) (13)
- 微软101道面试题 (1)
- 数理逻辑和数学基础 (92)
- 游戏编程 (6)
- 神经网络入门(连载) (6)
- 遗传算法入门(连载) (10)
- JAVA图形编程入门(连载) (9)
- 吴润昭私院 (1)
- 嘉兴新塍 (0)
- 党史陈列馆 (0)
- 我们的老家 (0)
- 程序设计自动化 (2)
- 算术算子的编译 (1)
- 4地址机器 (1)
- 53CM (1)
- 人工智能 (6)
- 二进制 (1)
- 十进制 (1)
- 轮盘赌 (1)
- 进化 (2)
- 变异 (1)

神经网络入门(连载之四)

标签：struct vector 工作 算法 网络

2006-01-10 11:54 26896人阅读 评论(16) 举报

分类：神经网络入门(连载) (5)

游戏编程中的人工智能技术
神经网络入门
(连载之四)

4.4 CNeuralNet.h（神经网络类的头文件）

在CNeuralNet.h文件中，我们定义了人工神经细胞的结构、定义了人工神经细胞的层的结构、以及人工神经网络本身的结构。首先我们来考察人工神经细胞的结构。

4.4.1 SNeuron（神经细胞的结构）

这是很简单的结构。人工神经细胞的结构中必须有一个正整数来纪录它有多少个输入，还需要有一个向量std::vector来表示它的权重。请记住，神经细胞的每一个输入都要有一个对应的权重。

```
Struct SNeuron
{
    // 进入神经细胞的输入个数
    int m_NumInputs;

    // 为每一输入提供的权重
    vector<double> m_vecWeight;

    //构造函数
    SNeuron(int NumInputs);
};
```

以下就是SNeuron结构体的构造函数形式：

```
SNeuron::SNeuron(int NumInputs): m_NumInputs(NumInputs+1)
(
    // 我们要为偏移值也附加一个权重，因此输入数目上要 +1
    for (int i=0; i<NumInputs+1; ++i)
    {
        // 把权重初始化为任意的值
        m_vecWeight.push_back(RandomClamped());
    }
}
```

- 旋转轮 (1)
- Roulette Wheel (0)
- 走迷宫 (1)
- 实无穷 (1)
- 潜在可实现性 (1)
- 马尔科夫<算法论> (1)
- 数理逻辑 (4)
- 数学基础 (1)
- 算法 (1)
- 游戏中的人工智能技术 (1)
- 勘误 (1)
- 清华大学出版社 (1)
- c语言 (1)
- pascal语言 (1)
- 集成开发环境 (1)
- 中文操作系统 (1)
- IDE (1)
- 分子图形 (7)
- MOL4D (4)
- 图形算法 (1)
- z缓冲法 (2)
- 画家算法 (2)
- 稳压器 (17)
- 稳压电源 (17)
- 负反馈 (1)
- 调整 (1)
- dosbox (1)
- dos程序 (1)
- dos编程 (1)
- Pascal线 (1)
- Steiner point (1)
- Plucker line (1)
- 计算机应用能力 (1)
- 上机考试 (1)
- 模拟软件 (1)
- 微波法检测 (1)
- 微波无损检测 (1)
- NDT (1)
- 微波应用 (1)
- 微波成像 (0)
- 何燮侯 (1)
- 何燮侯后代 (1)
- 免费电子图书 (1)
- 网上图书 (1)
- 外文书 (1)
- 科技书 (0)
- modern geometry (1)
- 现代几何 (1)
- 隐藏层 (1)
- 扫雷机 (1)
- 输入输出 (1)
- 神童·疯子 (1)
- 北医 (1)
- 赵炳文 (1)
- 同班同桌 (1)
- ATX开关电源电路图 (1)
- Godel完全定理 (1)
- Godel完备定理 (0)
- 哥德尔完全定理 (0)
- 哥德尔完备定理的证明 (0)
- Kirkman point (1)
- 北大 (1)
- 生物分类 (1)
- hp1000笔记本 (1)
- 分区 (1)

}
}
由上可以看出，构造函数把送进神经细胞的输入数目**NumInputs**作为一个变元，并为每个输入创建一个随机的权重。所有权重值在-1和1之间。

这是什么？我听见你在说。这里多出了一个权重！不错，我很高兴看到你能注意到这一点，因为这一个附加的权重十分重要。但要解释它为什么在那里，我必须更多地介绍一些数学知识。回忆一下你就能记得，激励值是所有输入*权重的乘积的总和，而神经细胞的输出值取决于这个激励值是否超过某个阈值(t)。这可以用如下的方程来表示：

$$w_1x_1 + w_2x_2 + w_3x_3 +...+ w_nx_n \geq t$$

上式是使细胞输出为 1 的条件。因为网络的所有权重需要不断演化（进化），如果阈值的数据也能一起演化，那将是非常重要的。要实现这一点不难，你使用一个简单的诡计就可以让阈值变成权重的形式。从上面的方程两边各减去t，得：

$$w_1x_1 + w_2x_2 + w_3x_3 +...+ w_nx_n - t \geq 0$$

这个方程可以再换用一种形式写出来，如下：

$$w_1x_1 + w_2x_2 + w_3x_3 +...+ w_nx_n + t * (-1) \geq 0$$

到此，我希望你已能看出，阈值t为什么可以想像成为始终乘以输入为 -1 的权重了。这个特殊的权重通常叫偏移（bias），这就是为什么每个神经细胞初始化时都要增加一个权重的理由。现在，当你演化一个网络时，你就不必再考虑阈值问题，因为它已被内建在权重向量中了。怎么样，想法不错吧？为了让你心中绝对敲定你所学到的新的人工神经细胞是什么样子，请再参看一下图12。

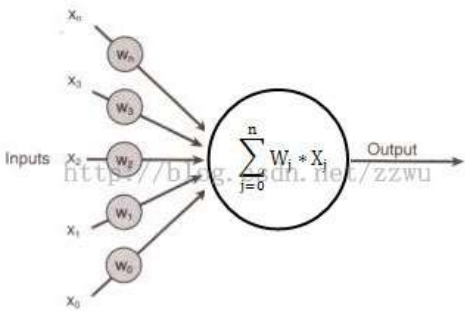


图12 带偏移的人工神经细胞。

4.4.2 SNeuronLayer（神经细胞层的结构）

神经细胞层SNeuronLayer的结构很简单；它定义了一个如图13中所示的由虚线包围的神经细胞SNeuron所组成的层。

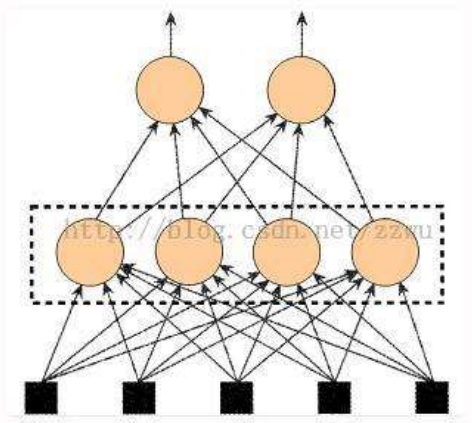


图13 一个神经细胞层。

- 重装系统 (1)
- win7 (1)
- CAI (1)
- CAI发展历史 (1)
- 马云 (2)
- 首富 (1)
- 传奇 (1)
- FDISK模拟 (1)
- Lehmer (1)
- FOCR (1)
- Deutche 困难问题 (1)
- QiQu Channel Router (1)
- 崎岖通道最优布线 (1)
- 达沃斯 (1)
- Pascal定理证明 (1)
- Pascal定理的Pacal证明原稿 (1)
- 键盘vk值 (1)
- LSI设计 (2)
- PCB设计 (2)
- FOCR的应用 (1)
- 快速最优通道布线 (1)
- 动画 (0)
- gif动画不能动 (0)
- RIBBONS (1)
- 蛋白质造型 (2)
- 顺势营养滴液 (1)
- JAVA小程序编程 (1)
- 苏联哲学百科 (2)
- 编译自动化 (1)
- VLSI布线 (1)
- PCB布线 (1)
- 计算机应用能力上机考试 (1)
- 博客分类目录 (1)
- MOL4DP (1)
- MOL4DE (1)
- zzwu博客 (1)
- 设计 (1)
- JAVA:First Contact (1)
- 磁芯测试仪 (1)
- Pascal Hexagrammum
Mysticum(六角迷魂图) (1)
- 马尔采夫定理 (1)
- 无穷长公式 (1)
- CSDN QA (1)
- 证明论 (1)
- 寻找计算机原理讲义 (1)
- 世界7大数学难题 (1)
- Circular points at infinity (1)
- 新浪博客怎样写 (1)
- Fields Medallists (1)
- 数学分支代码 (1)
- AI分支 (1)
- 喀秋莎 (1)
- 多值逻辑 (1)
- eknigu (1)
- 美国院士 (1)
- 国内外broks (1)
- Delphi应用程序的调试 (1)
- Delsphi的应用程序调试 (1)
- 分子图形软件介绍 (1)
- 我唱过的歌 (1)
- U盘存储原理 (1)
- 日文中的汉字 (1)
- 哲学系长寿多 (1)
- 100强企业 (1)

以下就是层的定义的源代码，它应该不再需要任何进一步的解释：

```
struct SNeuronLayer
{
    // 本层使用的神经细胞数目
    int          m_NumNeurons;

    // 神经细胞的层
    vector<SNeuron>  m_vecNeurons;

    SNeuronLayer(int NumNeurons, int NumInputsPerNeuron);
};
```

4.4.3 CNeuralNet（神经网络类）

这是创建神经网络对象的类。让我们来通读一下这一个类的定义：

```
class CNeuralNet
{
private:
    int          m_NumInputs;

    int          m_NumOutputs;

    int          m_NumHiddenLayers;

    int          m_NeuronsPerHiddenLyr;

    // 为每一层（包括输出层）存放所有神经细胞的存储器
    vector<SNeuronLayer> m_vecLayers;
```

所有private成员由其名称容易得到理解。需要由本类定义的就是输入的个数、输出的个数、隐藏层的数目、以及每个隐藏层中神经细胞的个数等几个参数。

```
public:

    CNeuralNet();

    该构造函数利用ini文件来初始化所有的Private成员变量，然后再调用CreateNet来创建网络。

    // 由SNeurons创建网络
    void  CreateNet();
```

我过一会儿马上就会告诉你这个函数的代码。

```
    // 从神经网络得到（读出）权重
    vector<double>  GetWeights()const;
```

由于网络的权重需要演化，所以必须创建一个方法来返回所有的权重。这些权重在网络中是以实数型向量形式表示的，我们将把这些实数表示的权重编码到一个基因组中。当我开始谈论本工程的遗传算法时，我将为您确切说明权重如何进行编码。

```
    // 返回网络的权重的总数
    int GetNumberOfWeights()const;

    // 用新的权重代替原有的权重
    void PutWeights(vector<double> &weights);
```

这一函数所做的工作与函数GetWeights所做的正好相反。当遗传算法执行完一代时，新一代的权重必须重新插入神经网络。为我们完成这一任务的是PutWeight方法。

```
    // S形响应曲线
```

- 中国上网费问题 (1)
- MSG200 (1)
- 中国上市公司价值排名100强名单 (1)
- 什么是椭圆几何与双曲线？ (1)
- 数理逻辑在计算机科学中的地位 (0)
- ACM Computing Classification System (1)
- 应对当前股市危机的八点建议 (1)
- Free Fortran Compilers (1)
- markdown (1)
- a (0)
- 最伟大的俄罗斯人 (1)
- 外公与大外公 (1)
- cmos工艺 (0)
- 著译目录 (1)
- 来函照登 (1)
- cmos工艺流程 (1)
- 一阶谓词逻辑的几个系统 (1)
- 潘多拉病毒 (1)
- 我的设计 (0)
- 一阶谓词逻辑系统的一个扩充 (1)
- 透视：一点透视 (0)
- 我的著译目录 (0)
- 参军65年：一些老照片 (0)
- 一些老照片 (0)
- 回忆类杂文 (0)
- 老照片 (0)
- 快速最优通道布线算法 (1)
- 中国学术界腐败 (0)
- 学术界腐败 (1)
- 扯下中国院士的神秘面纱(二) (1)
- 软件收费种类 (1)
- 抗战胜利日 (1)
- 50项世界顶级科技 (1)
- 国际设计自动化会议 (1)
- 计算机辅助设计年会 (1)
- 绍兴一中建校百年史 (1)
- 9.18 (1)
- 流亡三部曲 (1)
- 飲食寡淡是老年痴呆的禍根！ (1)
- desargues定理 (1)
- EKNIGU真棒！ (0)
- 简单因子和简单向 (0)
- 人工智能游戏编程 (0)
- 抗美援朝64周年纪念：老照片 (1)
- 基本形 (1)
- "一大"代表包惠僧de离奇经历 (1)
- 托派 (0)
- 刘仁静 (1)
- 中共一大代表李汉俊生平简介 (0)
- 能量与物质 (0)
- 抗战时期为何汉奸多 (0)
- 默克尔赠送中国桑叶地图 (0)
- 一大代表15人 (0)
- 能量的正负 (0)
- 负能量 (1)
- Puphotoshop用于照片整理 (0)
- 一大代表15人的图片处理 (1)
- 博客排名 (1)
- 李达在中共“一大”前后 (1)

inline double Sigmoid(double activation, double response);

当已知一个神经细胞的所有输入*重量的乘积之和时，这一方法将它送入到S形的激励函数。

// 根据一组输入，来计算输出
vector<double> Update(vector<double> &inputs);

对此Update函数函数我马上就会来进行注释的。

}; // 类定义结束

4.4.3.1 CNeuralNet::CreateNet（创建神经网络的方法）

我在前面没有对CNeuralNet的2个方法加以注释，这是因为我要为你显示它们的更完整的代码。这2个方法的第一个是网络创建方法CreateNet。它的工作就是把由细胞层SNeuronLayers所收集的神经细胞SNeurons聚在一起组成整个神经网络，代码为：

```
void CNeuralNet::CreateNet()
{
    // 创建网络的各个层
    if (m_NumHiddenLayers > 0)
    {
        //创建第一个隐藏层 [译注]
        m_vecLayers.push_back(SNeuronLayer(m_NeuronsPerHiddenLyr,
                                            m_NumInputs));

        for( int i=0; i<m_NumHiddenLayers-1; ++i)
        {
            m_vecLayers.push_back(SNeuronLayer(m_NeuronsPerHiddenLyr,
                                                m_NeuronsPerHiddenLyr));
        }

        [译注] 如果允许有多个隐藏层，则由接着for循环即能创建其余的隐藏层。

        // 创建输出层
        m_vecLayers.push_back(SNeuronLayer(m_NumOutput,m_NeuronsPerHiddenLyr));
    }

    else //无隐藏层时，只需创建输出层
    {
        // 创建输出层
        m_vecLayers.push_back(SNeuronLayer(m_NumOutputs, m_NumInputs));
    }
}
```

4.4.3.2 CNeuralNet::Update（神经网络的更新方法）

Update函数(更新函数)称得上是神经网络的“主要劳动力”了。这里，输入网络的数据input是以双精度向量std::vector的数据格式传递进来的。Update函数通过对每个层的循环来处理输入*权重的相乘与求和，再以所得的和数作为激励值，通过S形函数来计算出每个神经细胞的输出，正如我们前面最后几页中所讨论的那样。Update函数返回的也是一个双精度向量std::vector，它对应的就是人工神经网络的所有输出。

请你自己花两分钟或差不多的时间来熟悉一下如下的Update函数的代码，这能使你正确理解我们继续要讲的其他内容：

vector<double> CNeuralNet::Update(vector<double> &inputs)
{
// 保存从每一层产生的输出

托派分子 (0)

中国托派陈独秀 (1)

一大代表15人简单介绍 (1)

尼可斯基 (1)

一大代表15人的结局 (1)

歌德体下载 (1)

Manifest der Kommunistischen Partei (0)

一大代表竟有三个在黄埔军校当过官 (1)

初到俄罗斯 (0)

文章存档

2016年02月 (2)

2016年01月 (6)

2015年12月 (6)

2015年11月 (6)

2015年10月 (3)

展开

阅读排行

神经网络入门(连载之一) (97449)

神经网络入门(连载之二) (45246)

神经网络入门(连载之三) (34875)

神经网络入门(连载之四) (26882)

遗传算法入门(连载之一) (21753)

神经网络入门(连载之五) (21077)

遗传算法入门(连载之四) (18373)

神经网络入门(连载之六) (17881)

遗传算法入门(连载之二) (15456)

遗传算法入门(连载之十) (14698)

评论排行

神经网络入门(连载之一) (27)

遗传算法入门(连载之十) (26)

神经网络入门(连载之二) (24)

遗传算法入门(连载之一) (20)

神经网络入门(连载之四) (16)

神经网络入门(连载之三) (12)

遗传算法入门(连载之四) (11)

遗传算法入门(连载之五) (10)

遗传算法入门(连载之九) (10)

神经网络入门(连载之五) (7)

最新评论

我在CSDN论坛的提问与回复
zzwu: 1楼 密斯大白：您好！谢谢您的好意，我因身体原因，已不可能参加“CSDN老友记”答谢...
初到俄罗斯
zzwu: 物价那么贵，不知工资有多少？
神经网络入门(连载之四)
x249271850: // 计算权重*输入的乘积的总和。 netinput += m_vecLayers.m_vecNe...
神经网络入门(连载之四)
a_big_pig: 与9楼同问

```
vector<double> outputs;

int cWeight = 0;

// 首先检查输入的个数是否正确
if (inputs.size() != m_NumInputs)
{
    // 如果不正确，就返回一个空向量
    return outputs;
}

// 对每一层,...
for (int i=0; i<m_NumHiddenLayers+1; ++i)
{
    if (i>0)
    {
        inputs = outputs;
    }
    outputs.clear();

    cWeight = 0;

    // 对每个神经细胞,求输入*对应权重乘积之总和。并将总和抛给S形函数,以计算输出
    for (int j=0; j<m_vecLayers[i].m_NumNeurons; ++j)
    {
        double netinput = 0;

        int NumInputs = m_vecLayers[i].m_vecNeurons[j].m_NumInputs;

        // 对每一个权重
        for (int k=0; k<NumInputs-l; ++k)
        {
            // 计算权重*输入的乘积的总和。
            netinput += m_vecLayers[i].m_vecNeurons[j].m_vecWeight[k] *
                inputs[cWeight++];
        }

        // 加入偏移值
        netinput += m_vecLayers[i].m_vecNeurons[j].m_vecWeight[NumInputs-1] *
            CParams::dBias;

        别忘记每个神经细胞的权重向量的最后一个权重实际是偏移值，这我们已经说明过了，我们总是将它
        设置成为 -1的。我已经在ini文件中包含了偏移值，你可以围绕它来做文章，考察它对你创建的网路的功
        能有什么影响。不过，这个值通常是不应该改变的。

        // 每一层的输出，当我们产生了它们后，我们就要将它们保存起来。但用Σ累加在一起的
        // 激励总值首先要通过S形函数的过滤，才能得到输出
        outputs.push_back(Sigmoid(netinput,CParams::dActivationResponse)); cWeight = 0;
    }
}

return outputs;
}
```

遗传算法入门(连载之十) 神经网络入门(连载之四) a_big_pig: 3ks

一大代表15人的图片修理 zzwu: 欢迎大家指正和补充，特别是美术爱好者。

射影几何入门（连载一） - 1-1对baidu_33307963: 我的意思是如果两个无穷小 如果不是等比例缩小的话 是否还能建立一一映射关系？

射影几何入门（连载一） - 1-1对baidu_33307963: 我有一个地方始终想不清楚，希望能得到回复，无穷直线对应一条线段，我觉得如果从微分角度来考虑，是无法建...

神经网络入门(连载之四) evolone: 楼主大大，下载网址已经不能用了，能不能把工程的代码发给我一份，小弟毕设论文就是神经网络的题目，所以想...

我的博客（总目录） ershijiu: 向默默奉献的吴老师致敬！

上一篇

神经网络入门(连载之三)

下一篇

神经网络入门(连载之五)

我的同类文章

神经网络入门(连载)（5）

主题推荐

神经网络

猜你在找

查看评论

12楼 x249271850 2016-01-10 10:33发表



[cpp]

```
01. // 计算权重*输入的乘积的总和。
02.     netinput += m_vecLayers[i].m_vecNeurons[j].m_vecWeight[k] *
03.         inputs[cWeight++];
```

//楼主你好我想问，为什么对每个神经细胞他的权重还是一个数组呢？就是m_vecWeight[k]，不是每个神经细胞就一个权重么？

11楼 a_big_pig 2016-01-03 23:22发表



与9楼同问

10楼 evolone 2015-11-30 17:35发表



楼主大大，下载网址已经不能用了，能不能把工程的代码发给我一份，小弟毕设论文就是神经网络的题目，所以想好好学习研究一下，感谢！！
751788409@qq.com

9楼 gydgzd 2015-04-16 16:04发表



[cpp]

```
01. // 对每一个权重
02. for (int k=0; k<NumInputs-1; ++k)
03. {
04.     // 计算权重*输入的乘积的总和。
05.     netinput += m_vecLayers[i].m_vecNeurons[j].m_vecWeight[k] *
06.         inputs[cWeight++];
07. }
08.
09. // 加入偏移值
10. netinput += m_vecLayers[i].m_vecNeurons[j].m_vecWeight[NumInputs-1] *
11.     CParams::dBias;
```

在神经元的初始化中，偏移值被随机初始化在m_vecWeight中，为何此处还要专门有dBias这个变量，另外，既然偏移值已经在m_vecWeight中，最后一行其实可以省略吧，在上一个for循环中一起加

8楼 Lhfcws 2014-01-26 10:46发表



[cpp]

```
01. for( int i=0; i<m_NumHiddenLayers-1; ++i)
02. {
03.     m_vecLayers.push_back(SNeuronLayer(m_NeuronsPerHiddenLyr,
04.                                         m_NeuronsPerHiddenLyr));
05. }
```

SNeuronLayer构造的参数似乎有问题。

7楼 [zzwu](#) 2013-12-05 12:37发表



Special_Icy: 你好！

- 1.原书光盘中有全部源码；
- 2.我的个人网站[www@ggdn.net](#)本来有地方可下载，但现在此网站因我身体不好已不再经营维护，（没有去为域名和空间续费），所以现在已不能下载；
- 3.网上其他地方好像没有可下载源码的地方；
- 4.我准备开个免费网站或别的地方专门用来上载《游戏编程中的人工智能技术》一书的全部源码让大家下载，但要一些时间，现在还不能；
- 5.你要哪一部分源码，我可以找出来单独发给你。

Re: [pymqq](#) 2014-01-11 10:48发表



回复[zzwu](#)：<http://ishare.iask.sina.com.cn/download/explain.php?fileid=20574628> 这个地方可以下载源码

6楼 [向来痴从此醉](#) 2013-12-03 23:03发表



想问一下作者，源代码哪里能下载到呢？

Re: [pymqq](#) 2014-01-11 10:48发表



回复[向来痴从此醉](#)：这个地方可以下载：<http://ishare.iask.sina.com.cn/download/explain.php?fileid=20574628>

5楼 [zzwu](#) 2013-01-05 23:57发表



"激励总值首先要通过S形函数的过滤，才能得到输出",为什么非要经过S形函数（或者其他的函数）过滤呢？是为了让他收敛吗？

这仅仅是Mat先生的个人看法，你可以试试不通过S形函数的过滤，如何利用阶跃函数来直接控制扫雷机的各种动作。

4楼 [jack](#)- 2012-10-31 08:47发表



"激励总值首先要通过S形函数的过滤，才能得到输出",为什么非要经过S形函数（或者其他的函数）过滤呢？是为了让他收敛吗？

3楼 [jack](#)- 2012-10-31 08:30发表



如果输入一组数字，想得到这组数字的加权平均值，一旦通过了激励函数岂不是输出变成1了？？

2楼 [匿名用户](#) 2010-06-30 17:46发表



[e01]

1楼 [匿名用户](#) 2010-06-09 21:57发表



[e01]

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

- | | | | | | | | | | | |
|------------|---------------|----------|-----------|-----------|------------|----------------|-----------|---------|------------|-----------|
| 全部主题 | Hadoop | AWS | 移动游戏 | Java | Android | iOS | Swift | 智能硬件 | Docker | OpenStack |
| VPN | Spark | ERP | IE10 | Eclipse | CRM | JavaScript | 数据库 | Ubuntu | NFC | WAP |
| jQuery | BI | HTML5 | Spring | Apache | .NET | API | HTML | SDK | IIS | Fedora |
| XML | LBS | Unity | Splashtop | UML | components | Windows Mobile | Rails | QEMU | KDE | Cassandra |
| CloudStack | FTC | coremail | OPhone | CouchBase | 云计算 | iOS6 | Rackspace | Web App | SpringSide | Maemo |
| Compuware | 大数据 | aptech | Perl | Tornado | Ruby | Hibernate | ThinkPHP | HBase | Pure | Solr |
| Angular | Cloud Foundry | Redis | Scala | Django | Bootstrap | | | | | |

