信息熵

在介绍什么是信息熵前，先说说什么是熵。

熵是一个物理量，用于表示体系的混乱程度，熵越大，体系就越混乱。

而信息熵，就是用来衡量信息无序程度的量，同时它的值的大小，可用来衡量信息量的多少。这个概念由香农提出。

为什么信息熵可以用来衡量信息量的多少？

我们假设有一个样本，样本中的所有元素都是大写字母。

如果仅存在一种大写字母A，即A出现的概率为100%，那么这个样本仅有1种存在可能。

如果A和B出现的概率都为50%，那么随着元素量的增大，样本可能的表达就会有非常多种：譬如ABABAB……AABBAABB……

所以当一个事件出现的概率越小，其所含的信息量就越多。

信息量公式：

$$I_i = -log_2 P_i$$

信息熵则是信源所有可能事件的信息量的平均：

$$H(X) = -\sum_{j=1}^{n} P(x_j) \cdot log_2 P(x_j)$$

若仅有1件事件，概率为1，则信息熵为0。

所有事件等概率时，信息熵最大，若有N件等概率事件，信息熵为：

$$log_2 N$$

若编码器的平均码长等于信息熵，则这种编码就是最佳编码，若超过信息熵，则说明含有过剩的信息量，有冗余。

霍夫曼编码

假设我们收到了一段信息，信息由若干种符号组成，那么要怎么把这段信息转化为编码使其易于储存运输呢？

霍夫曼编码的算法是这样的：

①统计所有符号出现的概率（其实就是出现次数/总符号数）。

②将他们升序排序。

③把出现概率最小的两个符号合成一个新符号，新符号的出现概率就是二者的和。

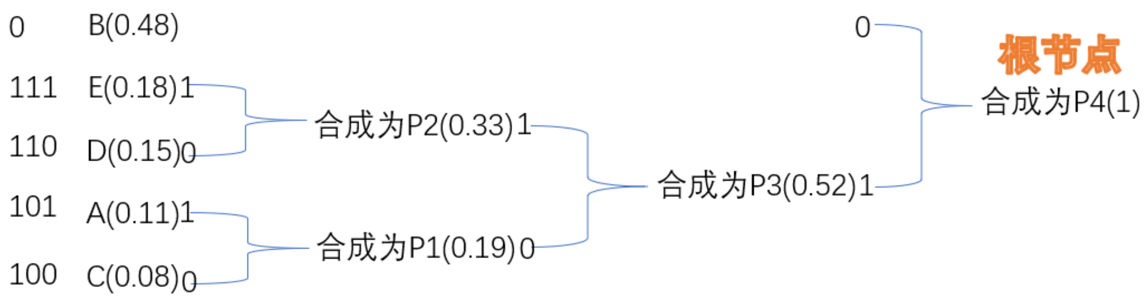
④把新符号丢进序列里，重复第②步。

⑤直到最后只剩一个符号时，这个合成的过程实际上就组成了一个二叉树，每个节点都是一个符号，然后都有左子树和右子树。

⑥接下来从根节点往下（不算根节点），比较两个子节点的出现概率的大小，然后根据某种规则给两个子节点赋值（比如大的给1，小的给0，或者相反），如果一样大随机赋值即可。

⑦最后一步步赋值到最初的单符号。每个符号的编码就是从根节点出发到这个符号过程中赋值的串。

为便于理解，请看下图：



大括号两端的数字是从根节点下搜时赋的值。

符号左侧就是该符号最终的编码。

这种编码方式使得最终二进制编码的长度（或者说大小）与其出现的概率成反比。

也就是出现概率越大，编码越短，出现概率越小，编码越长。

不过霍夫曼编码构造出的编码具有不唯一性（因为你下搜时赋值的规则是自己定的）。

且符号为等概率出现时，编码效率很低，而且错一个整串都得错。