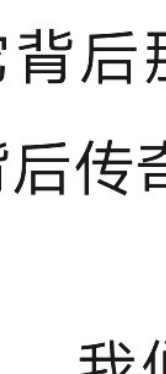


第151封信 | 通信中使用最多的算法——维特比算法



吴军



第151封信 | 通信中使用最多..

10:36

4.96MB



信件朗读者：宝木

小师弟，你好！

今天和明天我们说一个传奇的算法、它背后那个传奇的人，以及那个传奇的人背后传奇的公司。

我们要谈的这个人叫做安德鲁·维特比（Andrew Viterbi），你可能没有听说过他，因为通信行业之外的人知道他的并不多。不过你在媒体上应该经常看到对他所创建的高通公司（Qualcomm）的报道。

另外，通信行业的科技人员，如果不知道这个名字，要么赶快恶补一下通信技术的常识，要么就不用在这个行业混了，因为今天通信中使用最广泛的算法维特比算法（Viterbi Algorithm）就是他发明，并以他的名字命名的。

今天我们就先说说这个算法，这是我见到的最精妙的算法之一。它把现代数字通信中的解码复杂度降低了万亿倍，甚至更多。没有这个算法，你今天的手机打不了电话，你的计算机的硬盘无法工作。当然，什么语音识别，机器翻译等等，也无法完成了。

言归正传，下面我们就讲讲让作为科学家的维特比得以成名的这个算法。为了便于你理解算法，我还是用一个你比较熟悉的例子来解释，而不是维特比最初解决的通信问题。

经过我多次对语音识别的解释，你对此相对比较熟悉了，我们今天就从这里入手讲。

语音识别基本上分为三步，第一步是对声波进行信号处理，得到一些语音特征；第二步是通过语音特征，识别出相应的音节；第三步是根据音节，合成出完整的语句。我们今天就讲从第二步到第三步的过程。为了简化问题，我们假定第一步和第二步都是完美的，也就是说识别出的音节是100%正确的。当然在现实的各种识别系统中，能做到70%的单个音节识别率就非常好了。事实上我们人的识别率也就是这个水平，我们人之所以能够听清楚完整的句子，完全靠的是对上下文的理解，而非音节识别得准。

在汉语中，即使知道了读音，也就是我们小学学的拼音，要找到准确的对应的汉字还有一些困难，因为汉语中平均一个读音对应十几个汉字，即使你的四声读音非常准，耳朵也听得非常准（南方人其实常常做不到这一点），一个标上了四声声调的读音也对应了六个国标汉字。如果考虑到朱镕基的“镕”字还在国标里面，对应的汉字数量就更多了。我在写给你的第113封信中讲，汉语是一个二义性或者说歧义性非常强的语言，一个表现就在一音多字上。

一音多字这个问题，在语音识别上会产生指数爆炸的灾难性后果。比如一个音对应6个字，从理论上讲，两个音的组合就可能对应36个字的组合，一个长度只有十个字的短句子，10个拼音所对应的全部汉字串，能组合出6的十次方，也就是6000万种可能性。如果句子更长，组合数还会快速增长。

当然，可能有人会说考虑到一些字会组成常用的词，那么情况并不会这么糟糕。这种考虑在语音识别中的确是需要的，通过这种精简，可以大大减少排列组合的数量。但是，即使每一两个拼音，只对应少数几个单字词或者双字词，当句子稍微长一点，组合数量还是大得惊人。我们假定在考虑了组词之后每个拼音平均对应两个字，10个音节（拼音）可能的组合只有一千多种，比6000万种好了很多，但是，如果一句话的长度到了20个音节，可能的组合数量又达到了百万种。语音识别其实是一个任务，就是需要在这上百万的候选中找到最合理、最可能的语句。

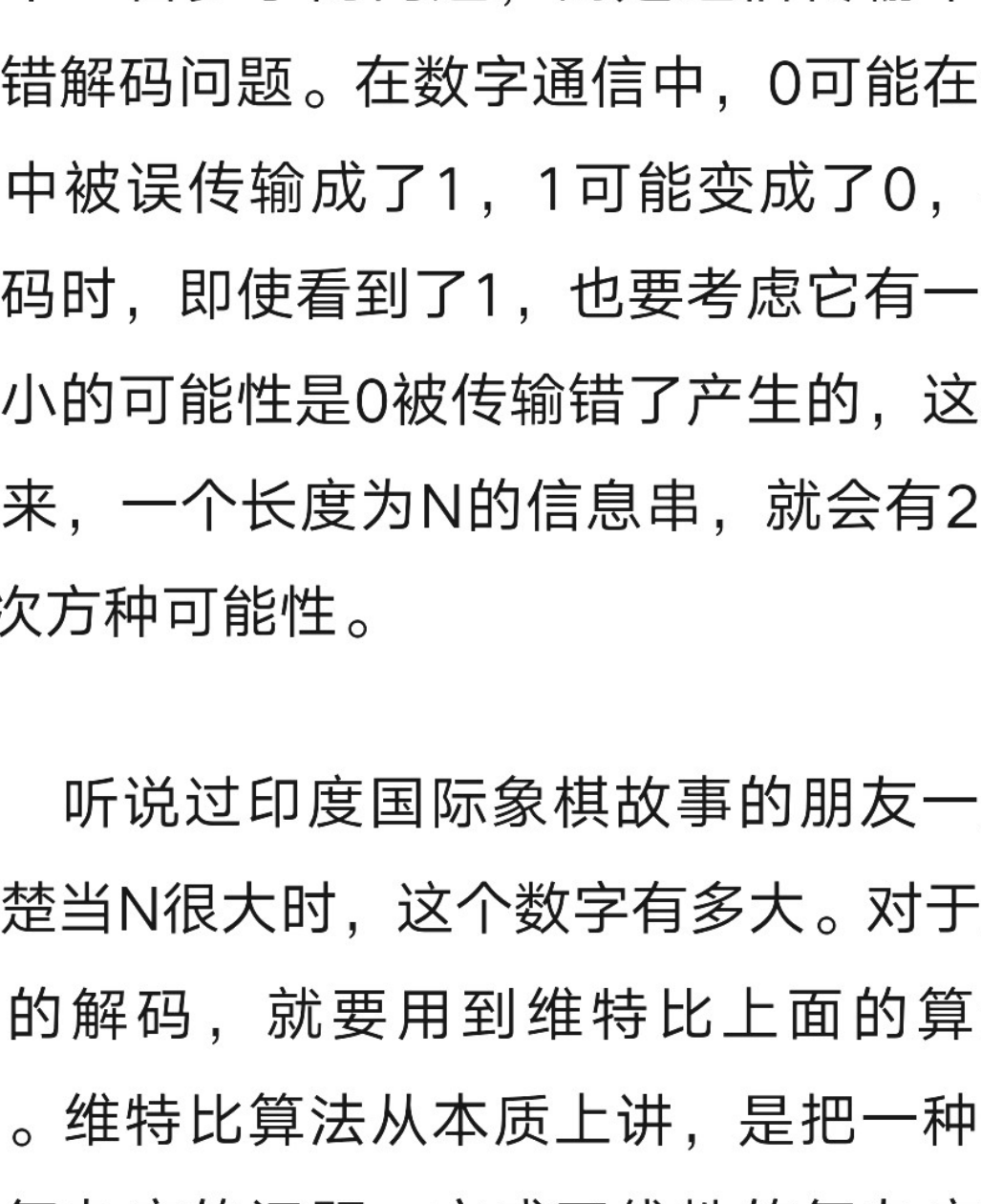
上述问题在通信中也会遇到，是一个标准的解码问题，那么怎么解决这个问题呢？将一百万种可能性一一评估显然不是一个好办法。对此维特比想到了一个特别妙的解码办法。为了比较清楚地说明问题，我把一句话的拼音和相应的汉字，做一个编号，按照下面的方式放到一张表中。

假如一个句子有20个音节，每个音节可能对应6个字，那么我们就用一个长为20，深度为6的网格将每个可能的字填入其中。

由于一个句子的音节按顺序从头到尾构成一个时间序列，因此，为了便于描述，我们用t表示音节的时间次序，将第一个音节标记为时间t = 1，最后一个音节标记为时间t = 20。

接下来，我们把每一个音节可以对应的汉字填入到相应的那一列的六个网格中，并且从1到6挨个标记。比如第一个音节对应的字，编号就是从W11，W12，到W16，第二个拼音对应的字，编号为W21，W22，到W26。编号中第一个数字是时间t，后一个数字是某个拼音所对应的六个候选汉字的序号。因此，到了最后一个拼音，即t = 20，相应的字就是W201，W202，到W206。在下面的表格中，我填入了第一个音和第六个音的情况。

所谓的解码就是在这样的网格中，找一条从t = 1到t = 20的路径，路径中经过的点，就是方格中的一个个字，比如W13，W24，到W201，等等。下图给出了一条路径。当然，对于一个20个音节的句子，这样的路径有三千六百万亿条。解码的过程就是找到一条最合理的路径的过程，我们把它称为寻找最佳路径。



对于这样路径密密麻麻的网格，维特比注意到这样一个现象：不论有多少条路径，最佳的那一条路径在某个特定时刻，只有六种可能。比如在t = 6（也就是第六个音节）的时刻，必须经过W61，W62，W63.....W66，这六个字中的一个。这样一来在第六个音节之前不论有多少条路径，真正有可能是最佳路径的候选路径，只可能有六条候选。

接下来，进入到第七个音节，由于这个音节也对应了6个汉字，和前面六条路径组合出6x6=36种可能性。但是，由于在第七个音节处，最佳的路径也只可能有六种（和t = 6的情况一样）。因此，我们只需要对第七个音节的六个候选，W71，W72，W73.....W76，每个候选保留一个最佳路径即可，这样一共也是六条。

以此类推，在每个时刻，我们只需要保留六条路径，将这个步骤一直走到t = 20最后的时刻。如果对于上述内容你一时难以理解，也没有关系，只要记住，在任何时刻，只要保留六条路径就好。

这种算法是维特比最早发明的，因此被称为维特比算法。对于20个音节，每个音节有6个候选字这种情况，维特比算法将复杂度从6的20次方，即3600万亿，减少到6x6x20=720次，足足减少了20万亿倍。即使考虑到采用了合理的组词，将每个拼音的候选字限制在两个，那么维特比算法也可以将复杂度从1百万次计算，减少到2x2x20=80次，减少了10000倍。

当然，当年维特比要解决的并不是汉语中一音多字的问题，而是通信传输中的纠错解码问题。在数字通信中，0可能在传输中被误传输成了1，1可能变成了0，在解码时，即使看到了1，也要考虑它有一个很小的可能性是0被传输错了产生的，这样一来，一个长度为N的信息串，就会有2的N次方种可能性。

听说过印度国际象棋故事的朋友一定清楚当N很大时，这个数字有多大。对于这样的解码，就要用到维特比上面的算法了。维特比算法从本质上讲，是把一种指数复杂度的问题，变成了线性的复杂度。这可能是所有的计算机算法中复杂度下降最大的改进。今天，除了语音识别、机器翻译、拼音转汉字、汉语的分词等用到了维特比算法，各种语言的拼写纠错，基因的测序，通信的解码都要用到它。

维特比是在上个世纪60年代发明了这个快速算法，并且凭借它，奠定了自己在数字通信中不可替代的地位。但是，维特比并不满足于停留在算法本身，而是努力将它推广出去。

Aa

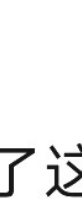
字号



写留言



8



请朋友读