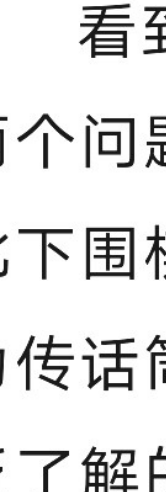


第120封信 | 打车软件匹配人和车是否比下围棋还复杂？



吴军

今天 00:00



第120封信 | 打车软件匹配人..

08:48 4.12MB



信件朗读者：宝木

小师弟，你好！

看到今天的标题，不知道你觉得上述两个问题哪一种更复杂？讲滴滴打车匹配比下围棋还复杂的是滴滴内部的高管，作为传话筒的是各种媒体和对计算机科学缺乏了解的人。讲到这里你已经猜出我对它们的评判了，滴滴打车的匹配远没有下围棋复杂。那么，为什么滴滴自己的人要说打车匹配更复杂呢？我们不妨听听他们主要的论据，他们是这样想的：

1. 围棋只有 $19 \times 19 = 361$ 个格子。

2. 如果把滴滴旗下的车作为棋盘上的行序列，把打车的人作为棋盘上的纵序列，那么从理论上讲，交叉点的数量可比361多多了。假如按照滴滴公布的数字，注册的出租车在千万辆的量级，打车的人在亿这个量级，组合起来可以是亿亿。

这种说法有道理么？没有。今天我们就来讲打车匹配的问题。在讲这个问题之前，我们先来看两个事实，从侧面说明打车匹配这个问题并没有想象的那么复杂。

1. 优步（Uber）的车在全球范围内可比滴滴一点也不小，但是，从来没有听优步的人讲匹配问题多么复杂。

2. 和打车匹配相似的问题很多，这类问题在数学上是等价的，也就是说，一个问题解决了，其他的也就都迎刃而解了。大家其实今天已经遇到很多这个问题的等价问题，我们不妨看一看：

第一个例子：Google的广告匹配问题。

Google搜索的关键词和相应的广告匹配也是一个多对多的问题。有人说把最相关的、出价最高的广告显示在搜索页不就行了？其实问题哪儿有那么简单，因为并非相关的广告就会被点击，而且每个广告主每天都有预算的，你不能在一天早上五分钟就把出价最高的广告显示完，这样晚上上网的人就看不到广告了。如果再考虑到每个人点击习惯的不同，那么广告的展示策略就很有讲究了。

第二个例子：Facebook的个性化广告。

你可以把每一个Facebook的用户看成一个打车人，每个广告商当作一辆车，这个匹配和打车匹配非常像。注意，由于Facebook的广告商也有预算，因此它可不是像电视台那样广播式的，把一种广告推给所有人，而是要一对一地推送，以便把几乎所有广告商的预算都花完。

第三个例子：交友网站的匹配。

你可以把男性看成打车人，女性看成出租车司机，那么他们的匹配也和打车匹配等价。

第四个例子：亚马逊的推荐。情况类似，就不多做分析了。

显然，你也没有听说上面任何一家公司说过匹配问题比下围棋难。那么这说明两个可能性必居其一：

第一个可能性，滴滴的人夸大其辞，以便吸引眼球；

第二个可能性，滴滴的人技术太差，搞不清楚下围棋的复杂度，也找不到自己问题的最佳算法。

至于哪个是真的，我们不做评论，我们接下来就来比比这两类问题的复杂度。

下围棋的问题比较好分析。它有361个格子，每个格子都有三种可能性，黑棋、白棋和空格。由于在每一个格子落子都是随意的，因此有3的361次方种可能性，大约是 10^{172} 。这个复杂度我在写给你的第一封信中分析过了，大得不得了。如果我们把宇宙中每一个基本粒子变成一个我们今天这么大的宇宙，这么多宇宙加起来的基本粒子的数量还没有下围棋的复杂度大呢。

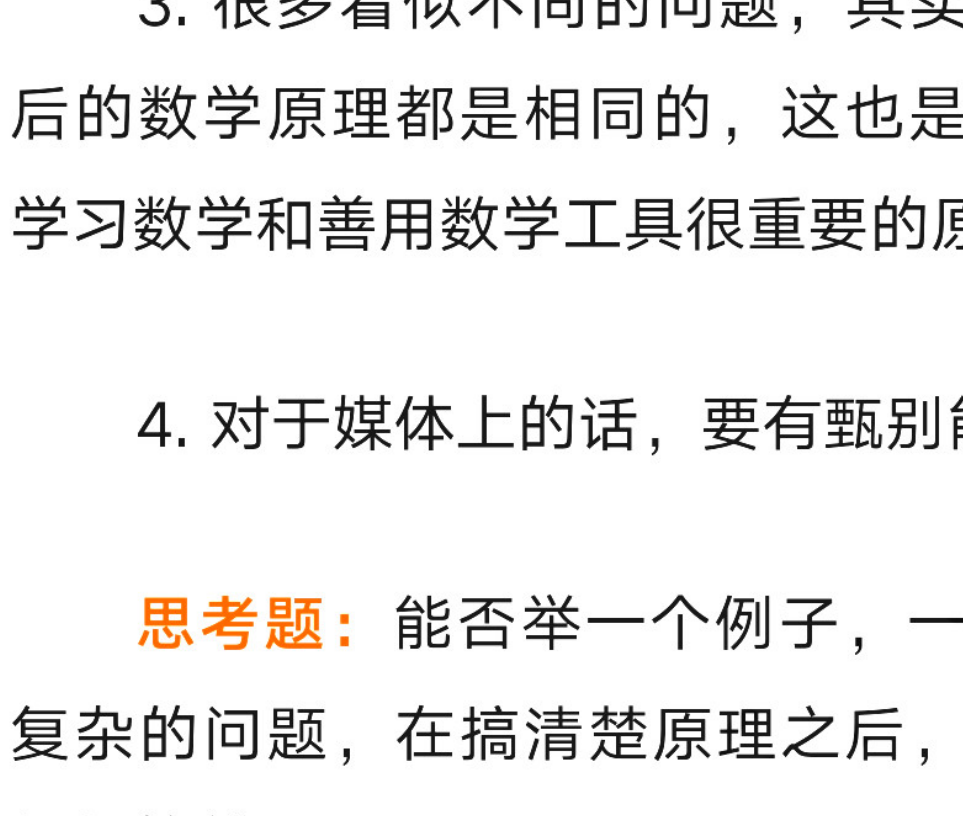
那么匹配问题的复杂度有多大呢？类似打车匹配，Google和Facebook的广告匹配，或者交友的匹配这种大类的问题，如果算法找得好，其实复杂度不是很大。

具体讲，如果整个网络中有M个人，N辆车，一一配对，复杂度只有M乘以N那么多，或者采用计算机科学严格的表述，就是 $O(M \times N)$ 。关于O的概念，在[写给你的第58封信](#)中，介绍算法的复杂度时，已经讲了，如果你忘了，可以复习一下。

如果是Google或者Facebook的广告系统，一个人看到的页面可以有多个广告，假如是K个广告，那么复杂度也不过是在M乘以N的基础上，再乘以一个K，也就是 $O(M \times N \times K)$ 。我们假定中国有1亿辆车同时在服务，全国10亿人都在路边等着打车，那么复杂度不过是十亿亿，即 10^{17} 次方，这个数字看上去不小，用1000台服务器也要算几个小时。但是如果换算成基本粒子中的中子或者质子，加起来一共不过100亿分之一公斤，或者0.1微克。0.1微克和近乎无限个宇宙，哪个复杂，便不言而喻了。事实上，不可能全国的人和车都在一个城市里。考虑到分散性之后，全中国所有打车匹配的计算量不超过几百亿，如果用1000台服务器算，是眨眼间就可以完成。

从这个例子可以看出，在计算机科学中，在人看来都很复杂的问题，但是那些问题真实的复杂度可以相差极大，近乎我们想象中的无穷大。当然，把一个看似复杂的问题变得不复杂的前提是找到好的算法。

关于上面这一类匹配问题，它们算法的基础要用到图论中一个二分图（bipartition graph）的概念，二分图其实并不复杂，你可以看一下下面的图。



在这个图中，所有的节点被分为了两类，红的是一类，蓝的是一类。红的和蓝的之间可以有连接，也就是它们可以做匹配，你可以把红节点看成打车的人，蓝节点看成是开车的司机。虽然上图中每一个红节点都和蓝节点有连接，但是在实际生活中，这个条件未必要满足，也就是说，一个点可能只和很少数量的对面的节点有连接。比如如果我们把接单的范围限制在5公里以内，尽管北京市有10万辆车在跑，和你能产生连接的可能只有七八辆。

在二分图中，红节点之间，蓝节点之间没有直接的关系。图中的每一条连线都可以有一个权重，通常在各个应用中，它们是设计出来的成本函数，或者利润函数。这个函数的设计就体现各个公司的领域知识了。

在二分图中，有一种被称为最大匹配的算法，它满足两个条件，**第一是保证节点数量少的一边所有的节点都有匹配，第二个是让整个网络中找到的匹配权重达到最大**。无论是滴滴打车的问题，还是Google、Facebook的广告展示问题，其实都是二分图中的最大匹配问题。而它的算法复杂度不高于M乘以N。这个算法最早是匈牙利数学家克尼格（Kőnig）证明的，当时还没有电子计算机。有兴趣的读者，可以去查一下这个算法。

最后总结一下今天的内容：

1. 很多看似同样复杂的问题，可能真实的复杂度相差甚远。**我们有时觉得一件事很复杂，是因为没有找到好办法。**

2. 对人来讲也是如此，两个看上去教育背景和经历差不多的人，本身的差距可能是数量级的，而见识上的差距就更大了。

3. 很多看似不同的问题，其实它们背后的数学原理都是相同的，这也是为什么学习数学和善用数学工具很重要的原因。

4. 对于媒体上的话，要有甄别能力。

思考题：能否举一个例子，一些看似复杂的问题，在搞清楚原理之后，其实变得很简单？

祝近安

吴军

2018年6月20日

Aa



写留言



65



请朋友读