



在如今这个互联网时代，有一家家喻户晓的公司，它自1998年问世以来，在极短的时间内就声誉鹊起，不仅超越了所有竞争对手，而且彻底改观了整个互联网的生态。这家公司就是当今互联网上的第一搜索引擎：谷歌 (Google)。

在这样一家显赫的公司背后，自然有许许多多商战故事，也有许许多多成功因素。但与普通商战故事不同的是，在谷歌的成功背后起着最关键作用的却是一个数学因素。

本文要谈的就是这个数学因素。

谷歌作为一个搜索引擎，它的核心功能顾名思义，就是网页搜索。说到搜索，我们都不陌生，因为那是凡地球人都会的技能。我们在字典里查个生字，在图书馆里找本图书，甚至在商店里寻一种商品等，都是搜索。如果我们稍稍推究一下的话，就会发现那些搜索之所以可能，并且人人都会，在很大程度上得益于以下三条：

1. 搜索对象的数量较小——比如一本字典收录的字通常只有一两万个，一家图书馆收录的不重复图书通常不超过几十万种，一家商店的商品通常不超过几万种等。

2. 搜索对象具有良好的分类或排序——比如字典里的字按拼音排序，图书馆里的图书按主题分类，商店里的商品按品种或用途分类等。

3. 搜索结果的重复度较低——比如字典里的同音字

通常不超过几十个，图书馆里的同名图书和商店里的同种商品通常也不超过几十种。

但互联网的鲜明特点却是以上三条无一满足。事实上，即便在谷歌问世之前，互联网上的网页总数就已超过了诸如图书馆藏书数量之类传统搜索对象的数目。而且这还只是冰山一角，因为与搜索图书时单纯的书名搜索不同，互联网上的搜索往往是对网页内容的直接搜索，这相当于将图书内的每一个字都变成了搜索对象，由此导致的数量才是真正惊人的，它不仅直接破坏了上述第一条，而且连带破坏了二、三两条。在互联网发展的早期，象 Yahoo 那样的门户网站曾试图为网页建立分类系统，但随着网页数量的激增，这种做法很快就“挂一漏万”了。而搜索结果的重复度更是以快得不能再快的速度走向失控。这其实是可以预料的，因为几乎所有网页都离不开几千个常用词，因此除非搜索生僻词，否则出现几十万、几百万、甚至几千万条搜索结果都是不足为奇的。

互联网的这些“不良特点”给搜索引擎的设计带来了极大的挑战。而在这些挑战之中，相对来说，对一、二两条的破坏是比较容易解决的，因为那主要是对搜索引擎的存储空间和计算能力提出了较高要求，只要有足够多的钱来买“装备”，这些还算是容易解决的。套用电视连续剧《蜗居》中

某贪官的台词来说，“能用钱解决的问题就不是大问题”。但对第三条的破坏却要了命了，因为无论搜索引擎的硬件如何强大，速度如何快捷，要是搜索结果有几百万条，那么任何用户想从其中“海选”出自己真正想要的东西都是几乎不可能的。这一点对早期搜索引擎来说可谓是致命伤，而且它不是用钱就能解决的问题。

这致命伤该如何治疗呢？药方其实很简单，那就是对搜索结果进行排序，把用户最有可能需要的网页排在最前面，以确保用户能很方便地找到它们。但问题是：网页的水平千差万别，用户的喜好更是万别千差，互联网上有一句流行语叫做：“在互联网上，没人知道你是一条狗”(On the Internet, nobody knows you're a dog)。连用户是人还是狗都“没人知道”，搜索引擎又怎能知道哪些搜索结果是用户最有可能需要的，并对它们进行排序呢？

在谷歌主导互联网搜索之前，多数搜索引擎采用的排序方法，是以被搜索词语在网页中的出现次数来决定排序，出现次数越多的网页排在越前面。这个判据不能说毫无道理，因为用户搜索一个词语，通常表明对该词语感兴趣。既然如此，那该词语在网页中的出现次数越多，就越有可能表示该网页是用户所需要的。可惜的是，这个貌似合理的方法实际上却行不大通。因为按照这种方法，任何一个像祥林嫂一样翻来复去倒腾某些关键词的网页，无论水平多烂，一旦被搜索到，都立刻会“金榜题名”，这简直就是广告及垃圾网页制造者的天堂。事实上，当时几乎没有一个搜索引擎不被“祥林嫂”们所困扰，其中最具讽刺意味的是：堪称互联网巨子的当年四大搜索引擎在搜索自己公司的名字时，居然只有一个能使之出现在搜索结果的前十名内，其余全被“祥林嫂”们挤跑了。

就是在这种情况下，1996年初，谷歌公司的创始人，当时还是美国斯坦福大学研究生的佩奇(Larry Page)和布林(Sergey Brin)开始了对网页排序问题的研究。这两位小伙子之所以研究网页排序问题，一来是导师的建议(佩奇后来称该建议为“我有生以来得到过的最好建议”)，二来则是因为他们对这一问题背后的数学产生了兴趣。

网页排序问题的背后有什么样的数学呢？这得从佩奇和布林看待这一问题的思路说起。在佩奇和布林看来，网页的排序是不能靠每个网页自己来标榜的，无论把关键词重复多少次，垃圾网页依然是垃圾网页。那么，究竟什么才是网页排序的可靠依据呢？出生于书香门第的佩奇和布林(两人的父亲都是大学教授)想到了学术界评判学术论文重要性的通用方法，那就是看论文的引用次数。在互联网上，与论文引用相类似的显然是网页链接。因此，佩奇和布林萌生了一个网页排序的思路，那就是通过研究网页间的相互链接来确定排序。具体地说，一个网页被其它网页链接得越多，它的

排序就越靠前。不仅如此，佩奇和布林还进一步提出，一个网页越是被排序靠前的网页所链接，它的排序就也应该越靠前。这一条的意义也是不言而喻的，就好比一篇论文被诺贝尔奖得主所引用，显然要比被普通研究者所引用更说明其价值。依照这个思路，网页排序问题就跟整个互联网的链接结构产生了关系，正是这一关系使它成为了一个不折不扣的数学问题。

思路虽然有了，具体计算却并非易事，因为按照这种思路，想要知道一个网页  $W_i$  的排序，不仅要知道有多少网页链接了它，而且还得知道那些网页各自的排序——因为来自排序靠前网页的链接更“值钱”。但作为互联网大家庭的一员， $W_i$  本身对其它网页的排序也是有贡献的，而且基于来自排序靠前网页的链接更“值钱”的原则，这种贡献与  $W_i$  的排序有关。这样一来，我们就陷入了一个“先有鸡还是先有蛋”的循环之中：想要知道  $W_i$  的排序，就得知道与它链接的其它网页的排序，而想要知道那些网页的排序，却又首先得知道  $W_i$  的排序。

为了打破这个循环，佩奇和布林采用了一个很巧妙的思路，即分析一个虚拟用户在互联网上的漫游过程。他们假定：虚拟用户一旦访问了一个网页后，下一步将有相同的几率访问被该网页所链接的任何一个其它网页。换句话说，如果网页  $W_i$  有  $N_i$  个对外链接，则虚拟用户在访问了  $W_i$  之后，下一步点击这些链接中任何一个的几率均为  $1/N_i$ 。初看起来，这一假设并不合理，因为任何用户都有偏好，怎么可能以相同的几率访问一个网页的所有链接呢？但如果我们考虑到佩奇和布林的虚拟用户实际上是对互联网上全体用户的一种平均意义上的代表，这条假设就不象初看起来那么不合理了。那么网页的排序由什么来决定呢？是由该用户在漫游了很长时间(理论上为无穷长时间)后访问各网页的几率分布来决定，访问几率越大的网页排序就越靠前。

为了将这一分析数学化，我们用  $p_i(n)$  表示虚拟用户在进行第  $n$  次浏览时访问网页  $W_i$  的几率。显然，上述假设可以表述为(请读者自行证明)：

$$p_i(n+1) = \sum_j p_j(n) p_{j \rightarrow i} / N_j$$

这里  $p_{j \rightarrow i}$  是一个描述互联网链接结构的指标函数(indicator function)，其定义是：如果网页  $W_j$  有链接指向网页  $W_i$ ，则  $p_{j \rightarrow i}$  取值为 1，反之则为 0。显然，这条假设所体现的正是前面提到的佩奇和布林的排序原则，因为右端求和式的存在表明与  $W_i$  有链接的所有网页  $W_j$  都对  $W_i$  的排名有贡献，而求和式中的每一项都正比于  $p_j$ ，则表明来自那些网页的贡献与它们的自身排序有关，自身排序越靠前(即  $p_j$  越大)，贡献就越大。



随机矩阵在谷歌算法里占据了重要的地位

为符号简洁起见，我们将虚拟用户第  $n$  次浏览时访问各网页的几率合并为一个列向量  $p_n$ ，它的第  $i$  个分量为  $p_i(n)$ ，并引进一个只与互联网结构有关的矩阵  $H$ ，它的第  $i$  行第  $j$  列的矩阵元为  $H_{ij} = p_{j-1}/N_j$ ，则上述公式可以改写为：

$$p_{n+1} = Hp_n$$

这就是计算网页排序的公式。

熟悉随机过程理论的读者想必看出来了，上述公式描述的是一种马尔可夫过程 (Markov process)，而且是最简单的一类，即所谓的平稳马尔可夫过程 (stationary Markov process)<sup>[1]</sup>，而  $H$  则是描述转移概率的所谓转移矩阵 (transition matrix)。不过普通马尔可夫过程中的转移矩阵通常是随机矩阵 (stochastic matrix)，即每一列的矩阵元之和都为 1 的矩阵（请读者想一想，这一特点的“物理意义”是什么？）<sup>[2]</sup>。而我们的矩阵  $H$  却可能有一些列是零向量，从而矩阵元之和为 0，它们对应于那些没有对外链接的网页，即

所谓的“悬挂网页” (dangling page)<sup>[3]</sup>。

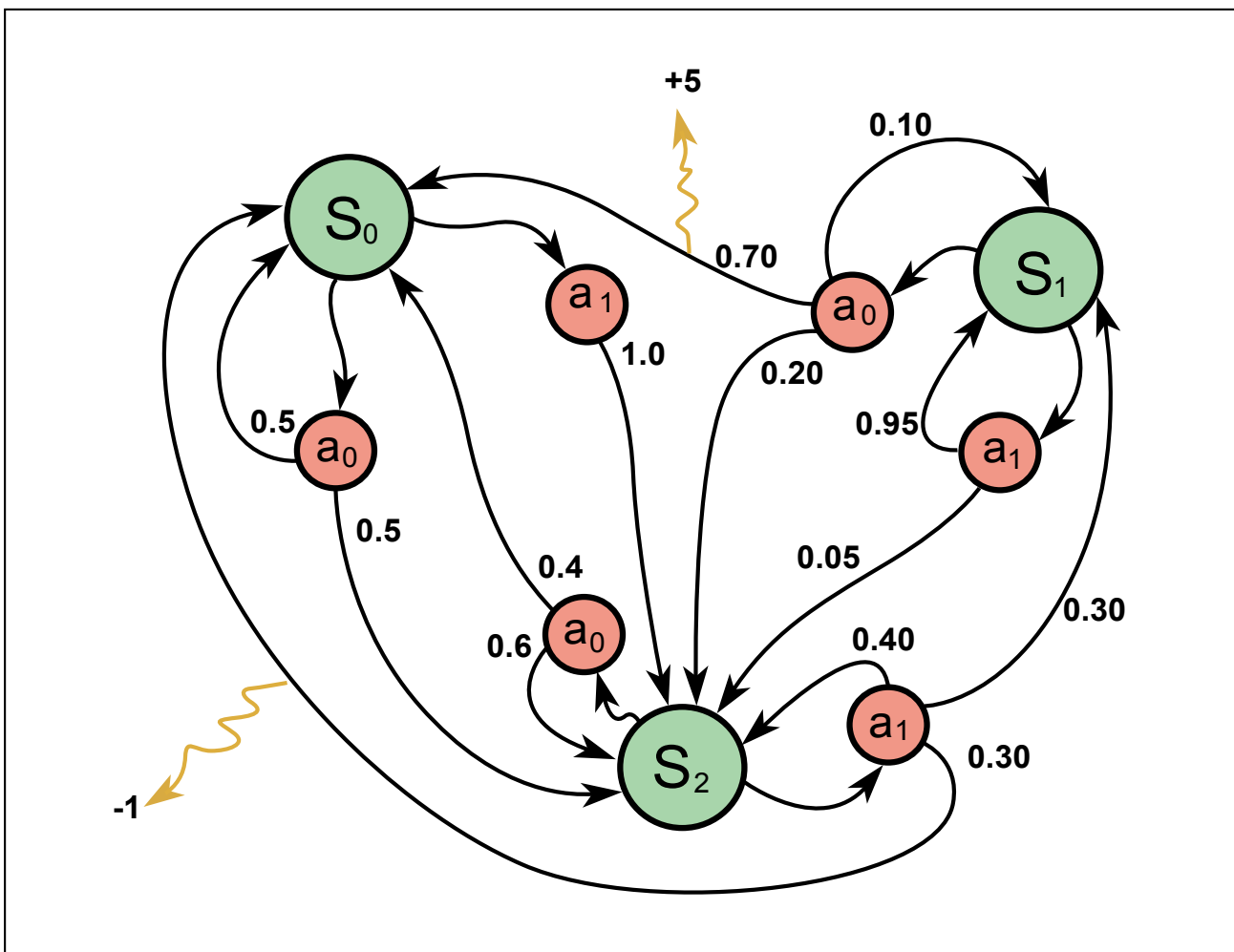
上述公式的求解是简单得不能再简单的事情，即：

$$p_n = H_n p_0$$

其中  $p_0$  为虚拟读者初次浏览时访问各网页的几率分布（在佩奇和布林的原始论文中，这一几率分布被假定为是均匀分布）。

如前所述，佩奇和布林是用虚拟用户在经过很长（理论上为无穷长）时间的漫游后访问各网页的几率分布，即  $\lim_{n \rightarrow \infty} p_n$ ，来确定网页排序的。这个定义要想管用，显然要解决三个问题：

1. 极限  $\lim_{n \rightarrow \infty} p_n$  是否存在？
2. 如果极限存在，它是否与  $p_0$  的选取无关？
3. 如果极限存在，并且与  $p_0$  的选取无关，它作为网页排序的依据是否真的合理？



马尔可夫过程 (Markov process) 在谷歌算法里占据了重要的地位。我们称离散的马尔可夫过程为马尔可夫链；其在各个时刻的状态转变由一个概率矩阵所控制。这是一个马尔可夫链的例子。

如果这三个问题的答案都是肯定的，那么网页排序问题就算解决了。反之，哪怕只有一个问题的答案是否定的，网页排序问题也就不能算是得到满意的解决。那么实际答案如何呢？很遗憾，是后一种，而且是其中最糟糕的情形，即三个问题的答案全都不是肯定的。这可以由一些简单的例子看出。比方说，在只包含两个相互链接网页的迷你型互联网上，如果  $p_0 = (1, 0)^T$ ，极限就不存在（因为几率分布将在  $(1, 0)^T$  和  $(0, 1)^T$  之间无穷振荡）。而存在几个互不连通（即互不链接）区域的互联网则会使极限——即便存在——与  $p_0$  的选取有关（因为把  $p_0$  选在不同区域内显然会导致不同极限）。至于极限存在，并且与  $p_0$  的选取无关时它作为网页排序的依据是否真的合理的问题，虽然不是数学问题，答案却也是否定的，因为任何一个“悬挂网页”都能象黑洞一样，把其它网页的几率“吸收”到自己身上（因为虚拟用户一旦进入那

样的网页，就会由于没有对外链接而永远停留在那里），这显然是不合理的。这种不合理效应是如此显著，以至于在一个连通性良好的互联网上，哪怕只有一个“悬挂网页”，也足以使整个互联网的网页排序失效，可谓是“一粒老鼠屎坏了一锅粥”。

为了解决这些问题，佩奇和布林对虚拟用户的行为进行了修正。首先，他们意识到无论真实用户还是虚拟用户，当他们访问到“悬挂网页”时，都不可能也不应该“在一棵树上吊死”，而是会自行访问其它网页。对于真实用户来说，自行访问的网页显然与各人的兴趣有关，但对于在平均意义上代表真实用户的虚拟用户来说，佩奇和布林假定它将会在整个互联网上随机选取一个网页进行访问。用数学语言来说，这相当于把  $H$  的列向量中所有的零向量都换成  $e/N$ （其中  $e$  是所有分量都为 1 的列向量， $N$  为互联网上的网页总数）。



百度的创始人李彦宏

如果我们引进一个描述“悬挂网页”的指标向量 (indicator vector)  $\mathbf{a}$ ，它的第  $i$  个分量的取值视  $W_i$  是否为“悬挂网页”而定，如果是“悬挂网页”，取值为 1，否则为零，并用  $\mathbf{S}$  表示修正后的矩阵，则：

$$\mathbf{S} = \mathbf{H} + \mathbf{a}\mathbf{e}^T/N$$

显然，这样定义的  $\mathbf{S}$  矩阵的每一列的矩阵元之和都是 1，从而是一个不折不扣的随机矩阵。这一修正因此被称为随机性修正 (stochasticity adjustment)。这一修正相当于剔除了“悬挂网页”，从而可以给上述第三个问题带来肯定回答（当然，这一回答没有绝对标准，可以不断改进）。不过，这一修正解决不了前两个问题。因此，佩奇和布林引进了第二个修正。他们假定，虚拟用户虽然是虚拟的，但多少也有一些“性格”，不会完全死板地只访问当前网页所提供的链接。具体地说，他们假定虚拟用户在每一步都有一个小于 1 的几率  $\alpha$  访问当前网页所提供的链接，同时却也有一个几率  $1-\alpha$  不受那些链接所限，随机访问互联网上的任何一个网站。用数学语言来说（请读者自行证明），这相当于是把上述  $\mathbf{S}$  矩阵变成了一个新的矩阵  $\mathbf{G}$ ：

$$\mathbf{G} = \alpha \mathbf{S} + (1-\alpha)\mathbf{e}\mathbf{e}^T/N$$

这个矩阵不仅是一个随机矩阵，而且由于第二项的加盟，它有了一个新的特点，即所有矩阵元都为正（请读者想一想，这一特点的“物理意义”是什么？），这样的矩阵是所谓的素矩阵 (primitive matrix)<sup>[4]</sup>。这一修正因此被称为素性修正 (primitivity adjustment)。

经过这两类修正，网页排序的计算方法就变成了：

$$\mathbf{p}_n = \mathbf{G}^n \mathbf{p}_0$$

这个算法能给上述问题提供肯定答案吗？是的，它能。因为随机过程理论中有一个所谓的马尔可夫链基本定理 (Fundamental Theorem of Markov Chains)，它表明在一个马尔可夫过程中，如果转移矩阵是素矩阵，那么上述前两个问题的答案就是肯定的。而随机性修正已经解决了上述第三个问题，因此所有问题就都解决了。如果我们用  $\mathbf{p}$  表示  $\mathbf{p}_n$  的极限<sup>[5]</sup>，则  $\mathbf{p}$  给出的就是整个互联网的网页排序——它的每一个分量就是相应网页的访问几率，几率越大，排序就越靠前。

这样，佩奇和布林就找到了一个不仅含义合理，而且数

学上严谨的网页排序算法，他们把这个算法称为 PageRank 不过要注意的是，虽然这个名称的直译恰好是“网页排序”。但它实际上指的是“佩奇排序”，因为其中的“Page”不是指网页，而是佩奇的名字。这个算法就是谷歌排序的数学基础，而其中的矩阵  $G$  则被称为谷歌矩阵 (Google matrix)。

细心的读者可能注意到了，我们还遗漏了一样东西，那就是谷歌矩阵中描述虚拟用户“性格”的那个  $\alpha$  参数。那个参数的数值是多少呢？从理论上讲，它应该来自于对真实用户平均行为的分析，不过实际上另有一个因素对它的选取产生了很大影响，那就是  $G^n p_0$  收敛于  $p$  的快慢程度。由于  $G$  是一个  $N \times N$  矩阵，而  $N$  为互联网上——确切地说是被谷歌所收录的——网页的总数，在谷歌成立之初为几千万，目前为几百亿，是一个极其巨大的数字。因此  $G$  是一个超大型矩阵，实际上是人类有史以来处理过的最庞大的矩阵。对于这样的矩阵， $G^n p_0$  收敛速度的快慢是关系到算法是否实用的重要因素，而这个因素恰恰与  $\alpha$  有关。可以证明， $\alpha$  越小， $G^n p_0$  的收敛速度就越快。但  $\alpha$  也不能太小，因为太小的话，“佩奇排序”中最精华的部分，即以网页间的彼此链接为基础的排序思路就被弱化了（因为这部分的贡献正比于  $\alpha$ ），这显然是得不偿失的。因此，在  $\alpha$  的选取上有很多折衷的考虑要做，佩奇和布林最终选择的数值是  $\alpha = 0.85$ 。

以上就是谷歌背后最重要的数学奥秘。与以往那种凭借关键词出现次数所作的排序不同，这种由所有网页的相互链接所确定的排序是不那么容易做假的，因为做假者再是把自己的网页吹得天花乱坠，如果没有真正吸引人的内容，别人不链接它，一切就还是枉然<sup>[6]</sup>。而且“佩奇排序”还有一个重要特点，那就是它只与互联网的结构有关，而与用户具体搜索的东西无关。这意味着排序计算可以单独进行，而无需在用户键入搜索指令后才临时进行。谷歌搜索的速度之所以快捷，在很大程度上得益于此。

在本文的最后，我们顺便介绍一点谷歌公司的历史。佩奇和布林对谷歌算法的研究由于需要收集和分析大量网页间的相互链接，从而离不开硬件支持。为此，早在研究阶段，他们就四处奔走，为自己的研究筹集资金和硬件。1998年9月，他们为自己的试验系统注册了公司——即如今大名鼎鼎的谷歌公司。但这些行为虽然近乎于创业，他们两人当时却并无长期从商的兴趣。1999年，当他们觉得打理公司干扰了自己的研究时，甚至萌生了卖掉公司的想法。他们的开价



谷歌公司创始人佩奇(左)和布林(右)

是一百万美元。与谷歌在短短几年之后的惊人身价相比，那简直就是“跳楼大甩卖”。可惜当时却无人识货。佩奇和布林在硅谷“叫卖”了一圈，连一个买家都没找到。被他们找过的公司包括了当时搜索业巨头之一的 Excite（该公司后来想必连肠子都悔青了）。为了不让自己的心血荒废，佩奇和布林只得将公司继续办了下去，一直办到今天，这就是谷歌的“发家史”。

谷歌成立之初跟其它一些“发迹于地下室”(one-man-in-basement)的IT公司一样寒酸：雇员只有一位（两位老板不算），工作场所则是一位朋友的车库。但它出类拔萃的排序算法很快为它赢得了声誉。公司成立仅仅三个月，《PC Magazine》杂志就把谷歌列为了年度最佳搜索引擎。2001年，佩奇为“佩奇排序”申请到了

专利，专利的发明人为佩奇，拥有者则是他和布林的母校斯坦福大学。2004年8月，谷歌成为了一家初始市值约17亿美元的上市公司。不仅公司高管在一夜间成为了亿万富翁，就连当初给过他们几十美元“赞助费”的某些同事和朋友也得到了足够终身养老所用的股票回报。作为公司摇篮的斯坦福大学则因拥有“佩奇排序”的专利而获得了180万股谷歌股票。2005年12月，斯坦福大学通过卖掉那些股票获得了3.36亿美元的巨额收益，成为美国高校因支持技术研发而获得的有史以来最巨额的收益之一<sup>[7]</sup>。

谷歌在短短数年间就横扫整个互联网，成为搜索引擎业的新一代霸主，佩奇和布林的那个排序算法无疑居功至伟，可以说，是数学成就了谷歌<sup>[补注]</sup>。当然，这么多年过去了，谷歌作为IT界研发能力最强的公司之一，它的网页排序方法早已有了巨大的改进，由当年单纯依靠“佩奇排序”演变成了由两百多种来自不同渠道的信息（其中包括与网页访问量有关的统计数据）综合而成的更加可靠的方法。而当年曾给佩奇和布林带来过启示的学术界，则反过来从谷歌的成功中借鉴了经验，如今一些学术机构对论文影响因子 (impact factor) 的计算已采用了类似“佩奇排序”的算法。

在本文的最后，还有一件事情在这里提一下，那就是与佩奇和布林研究排序算法几乎同时，有另外几人也相互独立地沿着类似的思路从事着研究<sup>[8]</sup>。他们中有一位是当时在美国新泽西州工作的中国人，他的算法后来也成就了一家公司——一家中国公司。此人的名字叫做李彦宏 (Robin Li)，他所成就的那家公司就是百度。这些新公司的发展极好地印证了培根 (Francis Bacon) 的一句名言：知识就是力量。

## 注释

1. 马尔可夫过程，也称为马尔可夫链 (Markov chain)，是一类离散随机过程，它的最大特点是每一步的概率分布都只与前一步有关。而平稳马尔可夫过程则是指转移概率与步数无关的马尔可夫过程（体现在我们的例子中，即  $H$  与  $n$  无关）。另外要说明的是，本文在表述上不同于佩奇和布林的原始论文，后者并未使用诸如“马尔可夫过程”或“马尔可夫链”那样的术语，也并未直接运用这一领域内的定理。
2. 在更细致的分类中，这种每一列的矩阵元之和都为 1 的随机矩阵称为左随机矩阵 (left stochastic matrix)，以区别于每一行的矩阵元之和都等于 1 的所谓右随机矩阵 (right stochastic matrix)。这两者在应用上基本是等价的，区别往往只在于约定。
3. 这种几乎满足随机矩阵条件，但有些列（或行）的矩阵元之和小于 1 的矩阵也有一个名称，叫做亚随机矩阵 (substochastic matrix)。
4. 确切地说，这种所有矩阵元都为正的矩阵不仅是素矩阵，而且还是所谓的正矩阵 (positive matrix)。这两者的区别是：正矩阵要求所有矩阵元都为正，而素矩阵只要求某个正整数次幂为正矩阵。
5. 读者们想必看出来， $p$  其实是矩阵  $G$  的本征值为 1 的本征向量，而利用虚拟用户确定网页排序的思路其实是在用迭代法解决上述本征值问题。在数学上可以证明，上述本征向量是唯一的，而且  $G$  的其它本征值  $\lambda$  全都满足  $|\lambda| < 1$ （更准确地说，是  $|\lambda| \leq \alpha$ ——这也正是  $G^n p_0$  的收敛速度与  $\alpha$  有关的原因）。
6. 当然，这绝不意味着在网页排序上已不可能再做假。相反，这种做假在互联网上依然比比皆是，比如许多广告或垃圾网页制造者用自动程序到各大论坛发帖，建立对自己网页的链接，以提高排序，就是一种常见的做假手法。为了遏制做假，谷歌采取了很多技术手段，并对有些做假网站采取了严厉的惩罚措施。这种惩罚（有时是误罚）对于某些靠互联网吃饭的公司有毁灭性的打击力。
7. 从投资角度讲，斯坦福大学显然是过早卖掉了股票，否则获利将更为丰厚。不过，这正是美国名校的一个可贵之处，它们虽擅长从支持技术研发中获利，却并不唯利是图。它们有自己的原则，那就是不能让商业利益干扰学术研究。为此，它们通常不愿长时间持有特定公司的股票，以免在无形中干扰与该公司存在竞争关系的学术研究的开展。
8. 那些研究与“佩奇排序”的类似仅仅在于大方向（即都利用互联网的链接结构来决定网页排序），而非具体算法类似。

## 参考文献

1. D. Austin, How Google Finds Your Needle in the Web's Haystack.
2. J. Battelle, The Birth of Google, Wired (August 2005).
3. S. Brin and L. Page, The Anatomy of a Large-Scale Hypertextual Web Search Engine, Seventh International World-Wide Web Conference, April 14-18, 1998, Brisbane, Australia.
4. O. Ibe, Markov Processes for Stochastic Modeling, (Elsevier Academic Press, 2009).
5. A. N. Langville and C. D. Meyer, Google's PageRank and Beyond: The Science of Search Engine Rankings, (Princeton University Press, 2006).
6. C. Rousseau and Y. Saint-Aubin, Mathematics and Technology, (Springer, 2008).

## 补注

有些读者对“是数学成就了谷歌”这一说法不以为然，认为是佩奇和布林的商业才能，或将数学与商业结合起来的才能成就了谷歌。这是一个见仁见智的问题，看法不同不足为奇。我之所以认为是数学成就了谷歌，是因为谷歌当年胜过其它搜索引擎的地方只有算法。除算法外，佩奇和布林当年并无其它胜过竞争对手的手段，包括商业手段。如果让他们去当其它几家搜索引擎公司的老总，用那几家公司的算法，他们是不可能脱颖而出的；而反过来，如果让其它几家搜索引擎公司的老总来管理谷歌，用谷歌的算法，我相信谷歌依然能超越对手。因此，虽然谷歌后来确实用过不少出色的商业手段（任何一家那样巨型的公司都必然有商业手段上的成功之处），而当年那个算法在今天的谷歌——如正文所述——则早已被更复杂的算法所取代，但我认为谷歌制胜的根基和根源在于那个算法，而非商业手段，因此我说“是数学成就了谷歌”。