# 盘古之白

为什么中文和 English、123 之间，要留一点空

王建硕

费曼式写法 · 由 AI 代理撰写与独立审校

# 导读

为什么中文和 English、123 之间，要留一点空

## 先看这句

“我用Mac写中文，今天提交了3个PR。”

这句话没有错。可你多看一秒，就会发现眼睛在“用Mac”和“了3个”那里轻轻绊了一下。于是很多人会改成：“我用 Mac 写中文，今天提交了 3 个 PR。”

这本小书讲的，就是那几处白。

它看起来像审美洁癖，其实背后有一条很实在的工程线：汉字、拉丁字母和数字的身体不同；活字排版早就把空白当材料；网页时代却常常只能让作者手动敲一个真实空格；pangu.js 这类工具把动作自动化；操作系统若也自动加空，又会碰到“是在帮我排版，还是在改我的文本”的边界。

> 大白话
>
> 贯穿全书的问题只有一个：中文和英文、数字之间需要的那点白，应该存在于眼睛里，还是存在于文件里？

读完它，你不一定会在每个地方都加空格。但你会知道什么时候该加、什么时候不该加，以及为什么一个小小的空格，会同时牵动字体、浏览器、输入法、复制粘贴和人的阅读疲劳。

# 第一章 · 两种身体挤在一行里

## 空格先不是规矩，是感觉

如果你写过这样的句子：今天上线了GPT-5，用户增长20%。你大概会在心里听见一个轻微的咯噔。

不是看不懂。中文读者当然能看懂。问题是，眼睛在扫过“了GPT-5”和“长20%”时，会短暂地停一下：这里到底从哪里开始是英文？数字跟前面的汉字有没有连在一起？这个停顿很小，小到你说不出它的名字；但一篇文章里出现几十次，它就变成疲劳。

所以很多人会写成：今天上线了 GPT-5，用户增长 20%。

中间那一点白，不是装饰。它像在人行道和车道之间画了一条浅浅的线，让两种速度不同的东西在同一条路上各走各的。

## 汉字和拉丁字母的身体不一样

字面，大白话说，就是一个字在排版时占住的那块地。汉字的传统身体接近正方形，一个字一个格子，排起来像一排砖。拉丁字母不是这样。一个 i 很窄，一个 W 很宽；同一个单词内部，字母彼此要靠得近，才能被眼睛当成一个词。

数字又是第三种东西。阿拉伯数字常常成串出现：2026、3.14、100%。它们不像汉字那样一个个独立，也不像英文单词那样靠拼写成词。它们更像小型仪表盘，读者会把它们整体抓出来。

当“中文English123中文”这样贴在一起时，问题不在任何一个字符本身。问题在边界。汉字的边界很清楚，英文和数字的边界靠连续性。如果两边硬贴，眼睛就要自己猜边界。

大白话

中文、英文、数字混在一行里时，空格做的不是“变好看”，而是“帮眼睛切块”。

## 为什么全角和半角会让人误会

全角，简单说，是一个字符大约占一个汉字宽度。半角，简单说，是英文、数字这类窄字符常用的宽度系统。很多中文排版争论会落到“全角半角”上，但这只能解释一半。

全角逗号和半角逗号的差别很明显，因为标点本身换了形状。可在“微信 Mac 版”里，Mac 还是 Mac，字母没有变成汉字宽。我们只是给它和汉字之间留了一口气。

这口气的大小也不是固定等于键盘上的空格。传统排版里理想的中西文间距，常常比一个英文空格小。你在浏览器里敲下的 ASCII 空格，是一个真实字符，会参与换行、复制、搜索、代码解析；它不是纯粹的视觉缝隙。

这就是后来所有工程问题的源头：我们想要的是视觉上的微小间距，但手里最容易拿到的工具，是一个会改变文本内容的空格字符。

## 空格其实是排版里的标点

中文没有天然的单词空格。英文几乎离不开单词空格。于是当两者混排时，那个夹在中间的空白就很尴尬：它不是中文标点，也不是英文单词之间的空格，却承担了两边的翻译工作。

它告诉读者：这里开始，阅读方式切换一下。前面按汉字的节奏读；后面把一串字母或数字整体读出来；再后面，回到汉字节奏。

一个好类比是门槛。门槛不是房间，也不是走廊，但没有门槛，两个空间就会糊在一起。中英文之间的空格就是门槛：它不表达词义，却表达边界。

## 这件小事为什么会吵起来

因为它同时属于审美和工程。

从审美看，加空格的人会觉得“不加就挤”。不加空格的人会觉得“多了一个洞”。两边都不是错觉。不同字体、字号、屏幕、行宽，会让同一个空格显得刚好或过宽。

从工程看，事情更麻烦。手动加空格，文本变干净了吗？在网页上可能更舒服；在搜索里可能多了一个字符；在代码名、品牌名、链接、文件路径旁边可能出错。你以为加的是白，其实加进了数据。

所以“为什么中文和英文、数字之间要加一个空格”这个问题，不能只回答“因为好看”。更准确的回答是：中文、英文和数字贴在一起时，读者需要边界；排版需要空白；而工程系统需要决定，这个空白到底是内容，还是显示。

后面几章，我们就沿着这条线走：先看空白在活字时代怎么被当成材料，再看网页时代为什么把它丢给了作者，最后看 pangu.js 和操作系统自动加空到底在解决什么，又制造了什么。

# 第二章 · 活字留下的呼吸

## 白不是空无，是一块材料

我们今天在键盘上按空格，会觉得自己只是“没打字”。但在活字排版里，空白不是没东西。它是一块真的东西。

活字，大白话说，就是每个字被做成一小块可以移动的字模。排版的人把这些小块按顺序排进版框，再上墨、压纸。字模之间要留空，不是靠想象，而是靠塞进不同厚度的空铅。

空铅，说白了，就是没有字面的金属小条或小块。它不印出墨迹，却决定墨迹之间隔多远。你看到纸上的白，是排字工人手里的一块“无字的字”。

这件事很重要。它提醒我们：排版不是把字摆上去就完了。排版是在安排黑和白。黑色是字，白色是字之间的路。

## 汉字排版先天有格子感

中文活字的基本节奏来自方块。每个汉字大体占一个正方形位置，字心有大有小，但字面给它一块稳定的地。于是中文正文的行看起来可以很整齐：一个字，一个步子。

这不是说所有汉字都一样宽，而是说中文排版长期形成了一种读法：字与字之间不靠空格切词，读者靠语义自己断开。白主要藏在字形内部、行距里、段落间，而不是每个词之间。

英文排版正相反。英文单词需要字母挨在一起，单词之间再用空格分开。太散，单词碎了；太紧，词和词粘了。英文的白，天然就站在词与词之间。

大白话

中文的默认节奏是“字字相接”；英文的默认节奏是“词词分开”。中英混排时，两个默认节奏撞上了。

## 混排真正难在“比例”

如果我们只把中英文之间塞一个键盘空格，很多时候会显得太宽。原因很简单：英文单词之间的空格，是为了隔开两个词；中英文之间的空白，只是为了标出脚步变化。它不需要那么大的门。

中文排版规范里常会讨论比一个汉字窄得多的间距，例如八分之一字宽这样的量级。字宽，大白话说，就是拿一个汉字的宽度当尺子。八分之一字宽，就是只在两个系统之间垫一点点，不是挖出一个明显的洞。

这也解释了为什么一些认真做字体和排版的人，并不喜欢手动 ASCII 空格。不是他们反对中西文留白，而是他们觉得这个空格太粗糙。它像用砖头垫桌脚：能垫，但不细。

然而在网页和普通编辑器里，作者常常没有“八分之一字宽”这个按钮。最容易、最稳定、最跨平台的办法，就是打一个普通空格。这是审美向工程妥协的第一步。

## 空白也有等级

我们可以把中文排版里的白分成几层。

- 字形内部的白：比如“田”和“口”里面的空。
- 字与字之间的白：多数时候很小，靠字体和渲染决定。
- 标点带来的白：逗号、句号、括号，会改变一行的呼吸。
- 中西文之间的白：它提示文字系统切换。
- 段落和版面上的白：它让读者知道哪里可以停下来。

初学排版的人容易只看字。熟练的人会看白。白一旦过多，页面松垮；白一旦过少，页面发闷。中英文之间的空格，就是这套白的秩序里最容易被普通写作者碰到的一块。

## 从铅空到代码

活字时代，空白的细度掌握在排字工和印刷系统手里。进入电脑以后，这个控制权被拆散了。

字体设计师决定字形边界。排版软件决定字符间距和换行。浏览器决定网页怎么渲染。作者决定文本里有没有空格。读者的系统再决定最后显示成什么样。

于是一个小空格不再只是审美判断。它变成多人协作里的接口问题：谁负责留白？字体负责，CSS 负责，输入法负责，还是作者负责？

这也是“盘古之白”这个名字的来处。神话里的盘古把混沌劈开；排版里的那一点白，把粘在一起的文字系统劈开。它很小，却让秩序出现。

下一章我们进入网页。到了那里，老师傅不在版台旁边了，页面被交给浏览器、CSS 和一堆默认规则。作者突然发现：自己必须用最笨的字符，去修最细的视觉问题。

# 第三章 · 网页没有老师傅

## 网页把排版拆成了几层

印刷品出厂时，排版已经定了。网页不一样。同一段 HTML，在手机、电脑、阅读器、不同浏览器、不同字体下，会长成不同的样子。

HTML，大白话说，是告诉浏览器“这里是什么内容”的标记。CSS，大白话说，是告诉浏览器“这些内容长什么样”的规则。理想情况下，文字内容应该交给 HTML，视觉间距应该交给 CSS。

按这个理想，中英文之间的那点白不应该由作者手敲。作者只写“中文English”，浏览器按照中文排版规则，在显示时自动垫一点视觉间距；复制出来仍然是原文，不多一个字符。这听起来很美。

现实是，这件事没有那么早成为稳定的浏览器能力。很长时间里，网页作者面对的选择粗糙得多：要么不加，忍受拥挤；要么手动加空格，让文本本身改变。

## 为什么浏览器不能“顺手做了”

浏览器当然会做很多排版判断，比如换行、字距、连字、标点挤压。那为什么不中西文自动加一点白？因为它看起来像小事，实际上牵动很多边界。

第一，语言边界不好判。一个页面可能有中文、日文、英文、代码、品牌名、邮箱、链接、数学表达式。浏览器如果一律加空，可能在不该断开的地方制造缝隙。

第二，空白大小不好定。印刷规范里说的是细间距，可能是一个汉字宽度的八分之一或类似量级。但网页里的字体各异，字号各异，粗细各异。固定值会在某些字体上刚好，在另一些字体上刺眼。

第三，兼容性很重。网页是一个承诺：旧页面今天也要能工作。浏览器如果突然改变混排间距，很多精确布局、按钮宽度、标题换行都会变。用户看到的是“浏览器升级后页面坏了”。

大白话

浏览器不是不知道空白好看，而是不敢轻易替所有网页决定：哪里加、加多少、算不算内容。

## CSS 想管的是“显示”，作者手里却是“字符”

排版系统最干净的分工应该是这样：文本存文本，样式管样式。中西文间距属于样式，因为它不改变句子的意思。

可普通写作者最容易控制的不是样式，而是字符。你在 Markdown、微博、博客后台、聊天窗口里，通常不能写复杂 CSS。你能做的，就是按一下空格。

Markdown，大白话说，是一种用普通文本写格式的办法。它鼓励人们用简单字符写文章，再转换成网页。Markdown 很适合写作，却也让“空格到底是内容还是样式”变得更模糊。因为空格看起来像排版动作，实际却进入了源文本。

这就是网页时代的尴尬：我们用内容层的工具，修显示层的问题。

## 手动空格的三个副作用

手动加空格能立刻改善阅读，但它不是免费午餐。

- **复制会带走它。**别人复制“微信 Mac 版”，得到的是带空格的字符串。多数时候没事，但放进命令、路径、产品名、搜索词里，有时就会变味。
- **换行会利用它。**普通空格是可断行位置。窄屏上，“Mac”可能被甩到下一行，视觉上未必比不加更好。
- **规则会不一致。**一个作者今天加，明天忘；一个团队里有人加英文，有人连数字也加，有人品牌名不加。文章越长，斑驳越明显。

这些副作用并不说明不该加。它们只说明：手动空格是一个工程折中，不是排版真理本身。

## 网页没有老师傅，就需要工具

在印刷厂里，经验丰富的排字工会替你照看这些细节。在网页里，发布链条变长了：作者写 Markdown，系统转 HTML，浏览器渲染，读者用不同设备打开。中间没有一个固定的老师

傅。

于是工具出现了。它们不试图理解全文意思，只做一件窄事：扫描字符串，找出汉字和半角英文、数字、符号贴住的地方，插入空格。

这就是 pangu.js 这类工具的出场位置。它不是美学大师，更像流水线上的小刀。它不会替你决定文章好不好，但能把最常见的混排粘连切开。

下一章我们拆这把小刀：它为什么能靠规则工作，它的边界在哪里，以及为什么“自动”并不等于“无脑”。

# 第四章 · pangu.js 的小刀

## pangu.js 做的事很窄

pangu.js 的目标可以用一句话说完：在中日韩文字和半角英文、数字、符号之间，自动插入适当空格，让混排更容易读。

中日韩文字，这里大白话说，就是汉字、日文假名、韩文等常见 CJK 字符。半角字符，这里主要指英文、数字和一些窄符号。pangu.js 不需要知道一篇文章讲什么，它只关心两类字符有没有贴在一起。

这也是它聪明的地方：问题被缩得很小。它不判断审美，不识别语法，不理解品牌策略。它只问：左边是不是 CJK？右边是不是 Latin 或数字？如果是，中间没有空格，那就补一个。

> 大白话
>
> pangu.js 像一把窄刀：不做饭，不摆盘，只负责把最常见的粘连切开。

## 规则为什么能工作

很多文字处理问题难，是因为机器要理解意思。中英文加空格相对容易，是因为大部分情况只看字符类别就够了。

比如“发布iOS17版本”。这里“布”和“i”贴住，“17”和“版”贴住。即使机器不知道 iOS 是什么，也能看出汉字和半角串的边界。处理后变成“发布 iOS 17 版本”，读者的眼睛轻松很多。

这类规则通常会围绕几组边界展开：

- CJK 字符和英文字母之间。
- CJK 字符和数字之间。
- CJK 字符和某些半角符号之间。
- 已经有空格或标点的地方，不重复添加。

真实实现会比这更细，因为括号、引号、斜杠、百分号、链接、代码片段都可能造成例外。但主干思想就是字符分类和边界替换。

## 它改的是文本，不只是显示

这里要特别小心。很多 pangu 系工具会直接返回一个带空格的新字符串。也就是说，它不是告诉浏览器“显示时稍微挪开”，而是真的把空格写进内容。

这有好处。结果稳定，任何浏览器、任何平台、任何复制粘贴之后都能看到空格。它也容易接入：博客发布前跑一遍，Markdown 保存前跑一遍，评论显示前跑一遍，都可以。

坏处也正来自这里。只要改文本，就会遇到“这里该不该改”的问题。代码里的变量名、URL、文件路径、邮箱地址、数学表达式、产品固定写法，都可能不希望被插入空格。

所以成熟的用法不会盲目扫全世界。它会限定范围：只处理正文节点，不处理代码块；只处理展示文本，不处理链接地址；只处理用户可读内容，不处理机器要解析的字段。

## 小工具背后的工程品味

pangu.js 让人着迷，不是因为它复杂，而是因为它克制。它承认自己解决的是一个小问题，然后把小问题做成可靠的管道。

幂等，大白话说，就是你对同一段文本处理一次和处理十次，结果应该一样。加空格工具很需要幂等。如果第一次把“中文English”变成“中文 English”，第二次不应该再变成“中文 English”。

正则表达式，大白话说，是一种用规则匹配字符串的办法。许多这类工具会用正则表达式识别边界。正则的好处是快、直接、容易移植；坏处是规则一多，就会变成一张难读的网。

工程品味就在这里：规则要够多，能覆盖常见场景；又不能多到没人敢维护。它不是写一条“全都加空”的命令，而是在“足够好”和“别误伤”之间反复调位置。

## 为什么叫盘古

这个名字好。盘古在混沌里开天辟地，而中英文数字粘在一起时，也像一个小混沌。工具跑过，黑字没有变多，意思没有变深，只是中间出现一线白，天地分开了。

但神话感不能遮住工程事实：pangu.js 不是标准，也不是唯一答案。它是一种作者侧、应用侧的补救办法。它存在，恰恰说明底层排版系统没有把这件事自然做好。

如果浏览器、编辑器、操作系统都能在显示层提供稳定、细腻、可关闭的中西文间距，很多场景就不需要把真实空格写进文本。可在那之前，小刀有小刀的价值。

下一章的问题也由此出现：如果工具能自动加空，操作系统和输入法要不要也自动加？它们是在帮用户，还是在擅自改用户的字？

# 第五章 · 自动加空的边界

## 自动加空为什么会惹人烦

如果一个系统在你输入“中文English”后，自动变成“中文 English”，很多人会觉得贴心。也会有很多人立刻生气。

生气的点不只是审美。真正的问题是：系统到底改了什么？如果它只是显示时拉开一点，用户复制出来还是原来的字符串，那它像排版功能。如果它把真实空格写进文本，那它就是替用户编辑内容。

显示层，大白话说，就是屏幕上看起来怎样。存储层，大白话说，就是文件、数据库、剪贴板里真正保存了什么。自动加空的争论，大多卡在这两层被混在一起。

> 大白话
>
> 用户想要的是看起来舒服；系统一不小心给的是内容被改写。

## 输入法不是排版软件

输入法的位置很特殊。它站在你和所有应用之间。你在聊天、写代码、填表、改文件名、输入密码、写论文，都可能经过输入法。

如果输入法默认替中英文加真实空格，它在写文章时可能很好，在写命令时可能危险，在输入产品型号时可能多余，在填数据库字段时可能破坏一致性。

举个小例子。“iPhone15”作为商品型号，商家可能希望它连在一起；“买了 iPhone 15”作为自然语言，读者希望它分开。输入法只看到一串字符，很难知道你现在是在写型号，还是在写句子。

所以系统级自动加空最需要的是克制：可关闭、可撤销、可按应用或文本区域区分。越靠近底层，越不能自信地替所有场景做决定。

## 视觉间距比真实空格更理想

从排版理想看，最好的办法不是到处插 ASCII 空格，而是在显示层增加细间距。这样文本仍然干净，排版也舒服。

例如，浏览器可以根据中文排版规则，在汉字和拉丁字母、数字之间渲染一个小于普通空格的间距。复制文本时不复制这段视觉白；搜索时不多一个字符；窄屏换行时也可以按排版引擎的规则处理。

这听起来像最终答案，但实现难度不低。它需要标准说明、浏览器支持、字体配合，还要让开发者能控制开关。否则同一个页面在不同浏览器里显示不同，作者还是会回到手动空格。

这也是为什么中文排版规范和 Web 实现之间常有落差。规范能说“这里应该有细间距”，但每天写字的人面对的是今天可用的编辑器和浏览器。

## 不同场景该用不同答案

把争论落到工程上，可以分四种场景。

- **面向人读的长文。**可以在发布前跑 pangu.js 这类工具，或由编辑手动整理。目标是稳定可读。
- **代码、命令、链接、数据字段。**不要自动插真实空格。这里字符就是协议，多一个空格可能改变含义。
- **浏览器和阅读器。**更适合做显示层细间距，让内容不变、视觉变好。
- **输入法和操作系统。**如果要做，应该让用户清楚知道它是在“排版建议”还是“改写文本”，并允许快速关闭。

你会发现，没有一个答案能覆盖所有地方。中文和英文之间的空格不是宗教问题，而是上下文问题。

## 那我们到底该不该加

如果你是在写给人读的中文文章，我的建议很简单：中文和英文、数字之间，通常加。尤其是长文、技术文章、产品文档、教程。它能减少眼睛的切换成本，也让页面显得更有秩序。

但你要记住自己加进去的是一个真实字符。遇到代码、URL、文件名、变量、固定品牌写法时，不要机械执行。排版规则服务阅读，不应该压过语义和协议。

更好的未来，是工具链分层：作者写干净文本，编辑器提供可见提示，发布系统按正文范围自动整理，浏览器逐步支持细腻的中西文间距，操作系统不擅自把显示偏好写进用户内容。

盘古之白，最美的地方不在“必须有一个空格”。它真正提醒我们的是：文字不是只有字符，阅读也不是只有意思。黑字之间那一点白，既是美学，也是接口。

把它留好，中文、English 和 123 才能在同一行里互不打架。

盘古之白 · 王建硕