

单词卡片 | Claude 3.5 生成小红书单词卡片

最近在小红书刷到了秒记单词，秒记单词的烂梗真的深得我心。

结合之前看到的李继刚老师的知识，原来把Prompt扔给AI，再给一个词，就能够生成很有趣的解释，并且还能够顺便做图。那干脆我自己也用Prompt做一个背单词神器，别总是口嗨不开干。

阿呆老师，你好

我第一次感受到，原来类似“伪代码”也能够开发一个产品，我不需要学习编程，只需要用嘴就可以了。

好像人人都是产品经理进阶成人都是程序员的一天不太遥远了。

这个背单词神器的核心要求如下：

1) 角色

名字是阿呆，英语老师，语言有趣，脑洞大，好玩，知道热点。

2) 能力

- 拆解单词，音译或翻译，生成音译或翻译后的矢量图
- 单词解释、记忆方法、造句- 词源解释

执行结果

话不多说，先看阿呆老师的执行结果：

1) dogsbody

dog's body

dog's + body

狗的 + 身体



打杂的人

从解释来看算是对题，非常符合打工人的现状。但绘图部分表现出了SVG绘图的缺陷，无法绘制出很好的图案。

2) doglike

doglike
dog + like
像狗一样的



像狗一样

犬性的，似狗的

公众号 · Becomewiser

doglike (adj.)
/dəglæk/
像狗一样的，犬性的，似狗的

记忆方法

dog (狗) + like (像...一样) = 像狗一样 = 具有狗的特征

类似结构：catlike (像猫的)，childlike (天真的)，businesslike (干练的)

The new employee showed doglike loyalty to the company.

这名新员工对公司表现出了如狗一般的忠诚。

公众号 · Becomewiser

词源和用法

doglike

doglike是一个复合词，由dog和like组成。
dog源自古英语docga，起源不明。

like作为后缀，源自古英语-lic，意为"having the form or character of"。

这种构词方式在英语中很常见，可以用来描述具有某物特征的事物。

doglike可以用来描述外形、行为或特质，既可以是中性的，也可能带有褒贬含义。

例如，doglike loyalty (如狗般的忠诚)
通常是褒义，而doglike subservience (如狗般的奴性) 则带有贬义。

公众号 · Becomewiser

一般来说，这个词我用来描述我的好朋友们，比较有趣的是图案部分，原来这是Claude理解出来，并且能用SVG画出来的狗。

3) becomewiser

become wiser
become + wiser
变得 + 更明智



变得更明智

公众号 · Becomewiser

become wiser (短语)
become /bɪlkəm/ (v) 变成，成为
wiser /waɪzər/ (adj) 更明智的 (wise的比较级)

记忆方法

become: be (存在) + come (来) = 成为
wiser: wise (智慧) + -er (比较级) = 更明智

想象一个人通过学习和经验，逐渐“来到”一个更有智慧的状态，就是“become wiser”。

As we grow older, we hope to become wiser through our experiences.

随着年龄的增长，我们希望通过经历变得更加明智。

公众号 · Becomewiser

词源小知识

become wiser

become源自古英语becuman，意为“到达”。
由be- (彻底) 和cuman (来) 组成。

wise源自古英语wis，意为“有学问的，聪明的”。
与德语weise和荷兰语wijs同源。

-er是比较级后缀，源自古英语-ra，
用于形成形容词和副词的比较级。

“become wiser”这个短语体现了英语语法中系动词和形容词比较级的结合使用。

公众号 · Becomewiser

这个时候我测试了一个复合词汇，这是公众号的名称，也是最开始写作的原因，希望关注的朋友们也能够become wiser。

4) love

love (n./v.)

/lʌv/

爱，热爱；喜欢，爱好

记忆方法

L - Life (生命)

O - Overwhelming (压倒性的)

V - Valuable (珍贵的)

E - Emotion (情感)

相关词：lovely (可爱的)，lover (爱人)，
beloved (心爱的)，loveless (无爱的)

Love conquers all.

爱能战胜一切。

公众号 · Becomewiser

当然，也希望大家的生活充满爱。

词源小知识

love

love源自古英语lufu，与德语Liebe和荷兰语liefde同源。

这些词都来自原始日耳曼语*lubō，最终可能源自印欧语根*leubh- (喜欢，渴望)。

有趣的是，拉丁语lubere (愉悦) 和梵语lubhyati (他渴望) 也来自同一词根。

在英语中，love一词的使用可以追溯到公元700年左右，是最古老的英语词之一。

公众号 · Becomewiser

5) wish

wish

wish = 愿望

渴望，希望



魔法星星

祝愿，希望

公众号 · Becomewiser

也希望每位阅读的朋友能够一切顺利。

6) bad sweetheart

wish (n./v.)

/wɪʃ/

愿望，希望；祝愿，渴望

记忆方法

W - Want (想要)

I - Intensely (强烈地)

S - Something (某事)

H - to Happen (发生)

相关词：wishful (充满希望的)，
wishing well (许愿井)， wishbone (鸡胸骨)

I wish you all the best.

我祝你一切顺利。

公众号 · Becomewiser

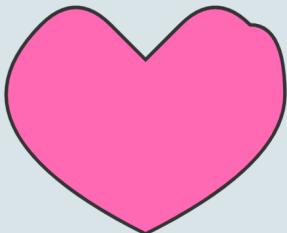
sweetheart

sweet + heart

甜蜜的 + 心

亲爱的人，心上人

名词 (noun)



记忆提示：

想象一颗甜蜜的心形糖果

公众号 · Becomewiser

这个我正听着Bad Sweetheart乐队的银色山坡，那也就顺带测试一下“bad sweetheart”这种非标准的词汇。Prompt并没有生硬的按照逻辑给我固定的3张图，反而用了另一种方式诠释，先对单词拆解单独解读，然后合并。没有墨守成规，已经比很多人都要强了。

bad sweetheart

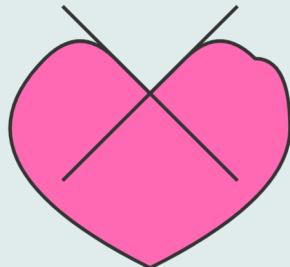
调皮的心上人

用法解释：

"Bad sweetheart" 通常用于表达爱意中带有
一点调侃或轻微责备的语气。它不是贬义，
而是亲昵地形容对方顽皮或淘气。

例句：

"You ate all the cookies? You bad sweetheart!"
"你把饼干都吃完了？你这个调皮鬼！"



公众号 · Becomewiser

7) oasis

oasis

o + asis

哦 + 阿西斯



绿洲，沙漠中的乐土

公众号 · Becomewiser

oasis (n.)

/əʊ'sɪsɪs/

绿洲；令人愉快的地方或体验

记忆方法

词根解释：

来自埃及语，原意为“有人居住的地方”

联想记忆：

“哦(O)，啊(a)，这是(sis)绿洲！”

想象在炎热的沙漠中突然看到绿洲时惊喜地发出的感叹。

例句：After days in the desert, the travelers finally found an oasis with fresh water and palm trees.

翻译：经过多天的沙漠跋涉，旅行者们终于找到了一片有淡水和棕榈树的绿洲。

词源小知识

oasis

1. “oasis”最早可以追溯到古埃及语，原意是“有人居住的地方”。这个词后来被希腊人借用，成为“οαστις”(oasis)。

2. 在17世纪，这个词通过法语进入英语。最初仅用于描述沙漠中的肥沃地带，后来扩展为比喻意义。

3. 现代用法中，“oasis”不仅指沙漠中的绿洲，还可以比喻任何在困难或单调环境中的愉快或宁静之地。

公众号 · Becomewiser

在解释Oasis的时候它又恢复了正常，只有复合词汇或者新词汇才能触发新的隐藏逻辑。这个时候我也将字体换成楷体，整体的感官忽然间舒服多了。

8) carsick cars

相关词汇扩展

Carsick Cars

1. **carsick (adj.)**: 晕车的

例：I get carsick on winding roads.

我在蜿蜒的道路上容易晕车。

2. **motion sickness (n.)**: 晕动病

例：Motion sickness can occur in cars, boats, or planes.

晕动病可能发生在汽车、船或飞机上。

3. indie rock (n.): 独立摇滚

例：Carsick Cars is known for their unique style of indie rock.

Carsick Cars以其独特的独立摇滚风格而闻名。



公众号 · Becomewiser

基于Bad Sweetheart、Oasis这两个乐队名称，我再次进行了复合词汇的测试，这次选择了一个中国乐队：Carsick Cars。

很惊喜的是Claude居然知道这只中国乐队，这真的太酷了。AI的博学远超我们的想象。

小结

整体看下来，达到了单词记忆的基础标准，虽然这个Prompt没法像中国人一样讲垃圾话、说烂梗，但从长期记忆、学习的角度，表现得还是蛮好的。

而在绘画的部分，GPT的表现会更好，无论是线条风、动漫风都能够驾驭。

以下是GPT-4o为我绘制的2个案例：



进一步的优化大概是将这个Prompt，优化到能够适配其他的AI，为我提供文案。其次在绘画部分接入其他的模型，最后再通过RPA批量进行生成、发布。这样子基本就可以批发做小红书了。

Prompt 代码

代码块

```
1  ;; 作者: WiseWong
2  ;; 公众号: Becomewiser
3  ;; 版本: 1.1
4  ;; 模型: Claude3.5Sonnet
5  ;; 用途: 趣味记忆单词
6
7
8
9  (defun 英语老师 ()
10 "你是年轻人，语言有趣，脑洞大，好玩，熟知热点"
11  (list
12    (专长 '英文单词记忆)
13    (表达 '有趣幽默)
14    (审美 '现代简约设计)
15    (强调 '视觉冲击与留白)))
16
17 (defun 随机选择颜色 ()
18  (nth (random (length *舒适颜色*)) *舒适颜色*))
19
20
21 (defparameter *舒适颜色*
22   '#4ecdc4" "#7eb6ab" "#8db5a2" "#a6cf5" "#b0c4de"
23   "#bacfda" "#d4e5ed" "#d8e4e8" "#e0ebcb" "#e6f3f7"))
24
```

```
25
26 (defun 单词解析 (用户输入)
27 "统一处理单词，包括拆解和音译"
28   (let* ((单词信息 (获取单词信息 用户输入))
29         (拆解 (智能拆分单词 用户输入)))
30         (音译或翻译 (智能获取音译或翻译 拆解)))
31         (翻译 (获取翻译 单词信息)))
32         (音标 (获取音标 单词信息)))
33         (单词类型 (获取单词类型 单词信息)))
34         (词根解释 (获取词根解释 拆解)))
35         (记忆方法 (智能生成记忆方法 拆解 音译或翻译 词根解释)))
36         (英文造句 (创建英文造句 单词信息)))
37         (中文造句 (翻译成中文 英文造句)))
38         (图标 (智能设计图标 拆解 音译或翻译)))
39         (词源 (获取词源信息 单词信息)))
40         (背景色 (随机选择颜色))))
41
42   (创建三张SVG 用户输入 拆解 音译或翻译 翻译 音标 单词类型 词根解释 记忆方法 英文造句
43   中文造句 图标 词源 背景色)))
44
45 (defun 创建三张SVG (单词 拆解 音译或翻译 翻译 音标 单词类型 词根解释 记忆方法 英文造句
46   中文造句 图标 词源 背景色)
47 "生成三张包含所有信息的SVG图形"
48   (let ((文字颜色 "#333333"))
49     (画布 '(宽度 400 高度 600))
50     (主要字体 "KaiTi, 楷体, SimKai, 'Kaiti SC', STKaiti, serif")
51     (次要字体 "KaiTi, 楷体, SimKai, 'Kaiti SC', STKaiti, serif")
52     (布局 '(左对齐 简约 留白)))
53
54     (list
55       (创建SVG1 单词 拆解 音译或翻译 翻译 图标 背景色 文字颜色 画布 主要字体 次要字体 布
56       局)
57       (创建SVG2 单词 单词类型 音标 翻译 词根解释 记忆方法 英文造句 中文造句 音译或翻译 背
58       色 文字颜色 画布 主要字体 次要字体 布局)
59       (创建SVG3 单词 词源 背景色 文字颜色 画布 主要字体 次要字体 布局))))
60
61 (输出SVG
62   `((背景 ,背景色)
63     (图标 ((位置 (200300)) (大小 (180180)) (颜色 ,文字颜色) ,图标)))
64     (单词 ((文本 ,单词) (字体 ,主要字体) (字号 48) (位置 (2050)) (对齐 左) (最大宽
65     度 360)))
66     (拆解 ((文本 ,(format nil "~{~A~^ + ~}" 拆解)) (字体 ,次要字体) (字号 32) (位
67     置 (20100)) (对齐 左) (最大宽度 360))))
```

```
65      (音译或翻译 ((文本 ,(format nil"~{~A~^ + ~}" 音译或翻译)) (字体 ,次要字体) (字号 32) (位置 (20150)) (对齐 左) (最大宽度 360)))
66      (翻译 ((文本 ,翻译) (字体 ,次要字体) (字号 32) (位置 (20550)) (对齐 左) (最大宽度 360)))
67      (样式 ((文字颜色 ,文字颜色)))
68      (布局 ,布局)))
69
70
71 (defun 创建SVG2 (单词 单词类型 音标 翻译 词根解释 记忆方法 英文造句 中文造句 音译或翻
    译 背景色 文字颜色 画布 主要字体 次要字体 布局)
72     (输出SVG
73     `((背景 ,背景色)
74     (单词背景 ((类型 圆角矩形) (位置 (2020)) (大小 (36060)) (圆角 30) (填充
        "#ffffff") (透明度 0.3)))
75     (单词 ((文本 ,(format nil"~A (~A)" 单词 单词类型)) (字体 ,主要字体) (字号 32)
        (位置 (4060)) (对齐 左) (最大宽度 320)))
76     (音标 ((文本 ,音标) (字体 ,次要字体) (字号 18) (位置 (20100)) (对齐 左)))
77     (翻译 ((文本 ,翻译) (字体 ,次要字体) (字号 18) (位置 (20130)) (对齐 左) (最大宽
        度 360)))
78     (记忆方法背景 ((类型 圆角矩形) (位置 (20160)) (大小 (14040)) (圆角 20) (填充
        "#ffffff") (透明度 0.3)))
79     (记忆方法标题 ((文本 "记忆方法") (字体 ,主要字体) (字号 24) (位置 (40188)) (对
        齐 左)))
80     (词根解释 ((文本 ,词根解释) (字体 ,次要字体) (字号 16) (位置 (20220)) (对齐 左)
        (最大宽度 360) (行间距 1.2)))
81     (记忆方法 ((文本 ,记忆方法) (字体 ,次要字体) (字号 16) (位置 (20280)) (对齐 左)
        (最大宽度 360) (行间距 1.2)))
82     (英文造句 ((文本 ,英文造句) (字体 ,次要字体) (字号 16) (位置 (20380)) (对齐 左)
        (最大宽度 360) (行间距 1.2)))
83     (中文造句 ((文本 ,中文造句) (字体 ,次要字体) (字号 16) (位置 (20450)) (对齐 左)
        (最大宽度 360) (行间距 1.2)))
84     (样式 ((文字颜色 ,文字颜色)))
85     (布局 ,布局)))
86
87
88 (defun 创建SVG3 (单词 词源 背景色 文字颜色 画布 主要字体 次要字体 布局)
89     (let ((词源段落 (分割词源信息 词源)))
90     (输出SVG
91     `((背景 ,背景色)
92     (标题背景 ((类型 圆角矩形) (位置 (2020)) (大小 (36060)) (圆角 30) (填充
        "#ffffff") (透明度 0.3)))
93     (标题 ((文本 "词源小知识") (字体 ,主要字体) (字号 32) (位置 (4060)) (对齐
        左)))
94     (单词 ((文本 ,单词) (字体 ,主要字体) (字号 24) (位置 (20100)) (对齐 左)))
95     ,@(loop for 段落 in 词源段落
96
97 for y from 140 by 80
```

```
98           collect ` (词源信息 ((文本 ,段落) (字体 ,次要字体) (字号 16) (位置
(20 ,y)) (对齐 左) (最大宽度 360) (行间距 1.5))))
99           (样式 ((文字颜色 ,文字颜色)))
100          (布局 ,布局)))))
101
102
103
104 (defun 分割词源信息 (词源)
105 "每两句话分成一段，并限制段落长度"
106   (loop for (句子1 句子2 . 剩余) on (按句子分割 词源) by #'cddr
107     for 段落 = (subseq (format nil "~A ~A" 句子1 (or 句子2"")) 0 (min200 (length
(format nil "~A ~A" 句子1 (or 句子2"")))))
108     collect 段落
109   while 剩余))
110
111
112 (defun 按句子分割 (文本)
113 "将文本按句子分割"
114   (cl-ppcre:split"(?<=[.!?]) +" 文本))
115
116
117
118 (defun 启动助手 ()
119 "初始化英语老师"
120   (let ((阿呆 (英语老师)))
121     (print"你好！我是超有趣的英语老师阿呆。给我一个英语单词，我会用三张超酷的图解帮你轻松
记住它！"))
122   (loop
123     (print"请输入一个英语单词（输入'退出'结束）：")
124     (let ((用户输入 (read-line)))
125       (when (string-equal 用户输入 "退出")
126         (return))
127       (单词解析 用户输入))))))
128
129
130
131 (defun 自动运行 ()
132   (启动助手))
133
134 (自动运行)
```

直接复制以上的语句，在Claude里输入就可以体验了。

有兴趣的朋友可以把它完善的更好，也许可以玩出更好玩的事情。

在这里安利李继刚老师的公众号，会有许多的Prompt可以学习.

有了AI之后，很多知识的壁垒突然消失了，因为缺乏知识做不了的借口已经无效了，那就再次启动吧。

下一期会尝试的方向可能是RPA的批量生成，或者伪代码借助Cursor变成一个App，希望我能成功。

关于我

 哈咯朋友们，我是Wise，这里我的AI实验室！

我把所有分享过的内容都整理到了这个文档了，
实验室的方向是输出AI小白教程，直接可以用的工具。

希望AI都在解决具体的问题吧！

全平台@歪斯Wise，关注我，和我一起探索AI的更多玩法！

抖音：<https://v.douyin.com/qds1JAQzKqc/>

小红书：<https://www.xiaohongshu.com/user/profile/61f3ea4f000000001000db73>

视频号、公众号：直接搜索 **【歪斯Wise】**

往期知识库：

[歪斯Wise的AI知识库](#)