

6/5

Random Process (隨機程序)

Random Number Generator

- `from random import randint, random, choice, sample`
- `randint(a, b)`
 - random integer in range [a, b]
- `random()`
 - x in the interval [0, 1)
- `choice(['paper', 'scissor', 'stone'])`
 - Choose a random element from a list
- `sample(['鼠', '牛', '虎', '兔', '龍', '蛇', '馬', '羊', '猴', '雞', '狗', '豬'], 3)`
 - Chooses k unique random elements from a list

Text Generator

唬爛產生器

繳交報告、漢字數、應付男/女朋友的好夥伴

* 請輸入您的主題名稱

下大雨

* 請輸入字數要求(上限1000)

50

產生

趙樹理說過一句富有哲理的話，讀書也像開礦一樣“沙裡淘金”。但願諸位理解後能從中有所成長。我們需要淘汰舊有的觀念，奧斯特洛夫斯基講過一句值得人反覆尋思的話，我只相信一條：靈感是在勞動的時候產生的。這段話看似複雜，其中的邏輯思路卻清晰可見。現在，正視下大雨的問題，是非常非常重要的。因為，若能夠欣賞到下大雨的美，相信我們一定會對下大雨改觀。

Developer
徐子修 Bill Hsu

Contact

藏頭詩



ChatGPT

請幫我產生一首七言絕句，開頭四個字分別是「高」、「雄」、「女」、「中」。

高山流水韻悠揚，
雄鷹展翅破雲霜。
女兒心事誰能解，
中窻獨坐思斷腸。

Message ChatGPT

ChatGPT can make mistakes. Check important info.

人工智慧 (Artificial Intelligence)

- [Webster] The capability of a machine to imitate intelligent human behavior.
 - 韋氏詞典: 讓機器具備**模仿**人類智慧行為的能力。

- 它只需要表現得有彷彿有智慧。它不必真的具備採取那個行為所需要的智慧。

具體的例子

- 我們早就每天都在用 - 注音輸入法
- 台北是
- 台北市民

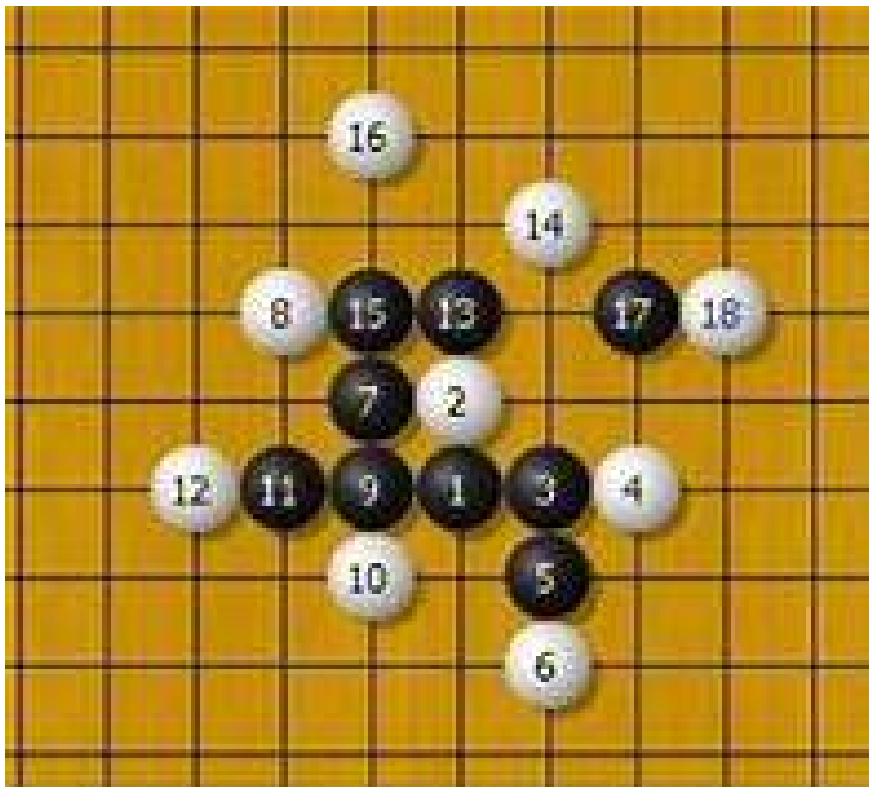
電腦如何選字？

1. 統計
2. 文法

很少見到「台北釋」，
但常見「台東釋迦」

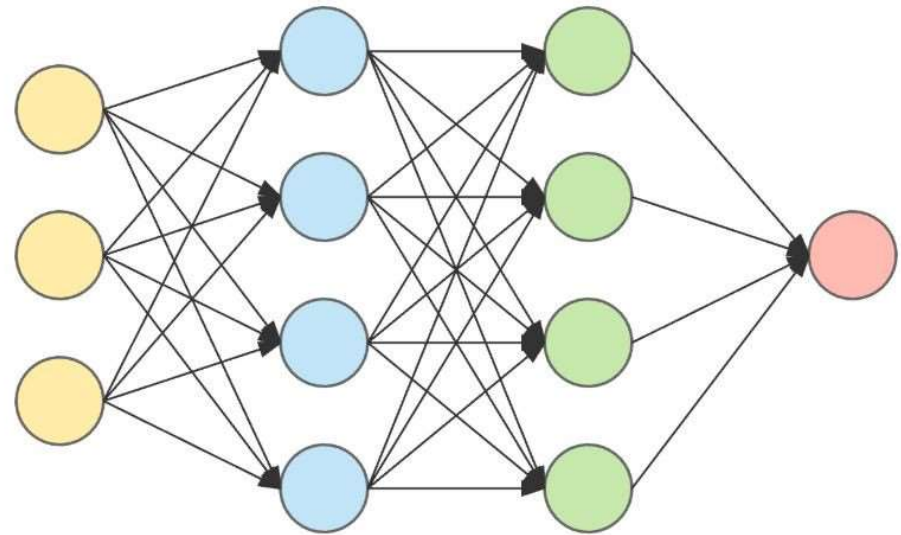
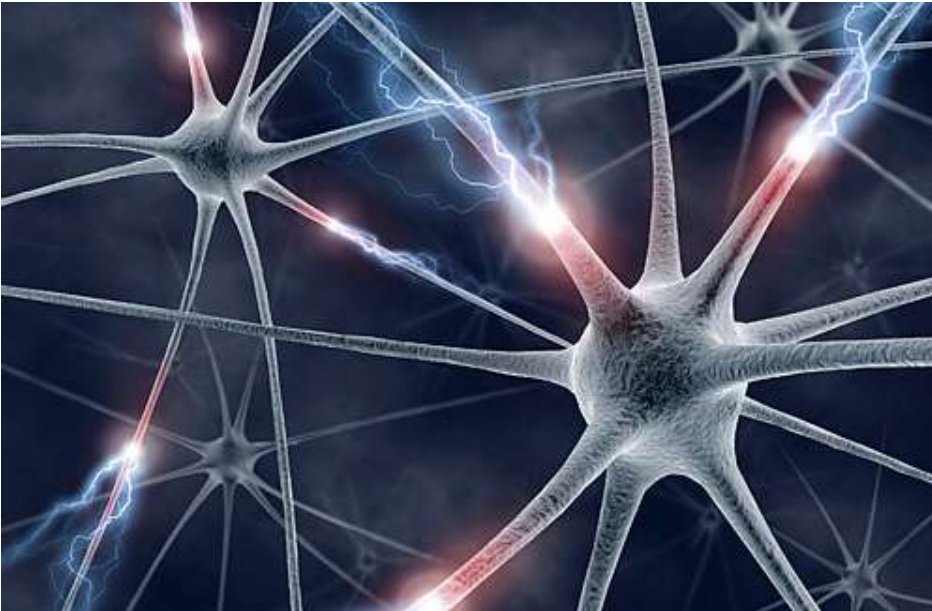
下棋的策略

- 最簡單的策略：熟背上萬部棋譜，直接根據對手下的位置來回應



熟讀唐詩三百首，
不會做詩也會吟。

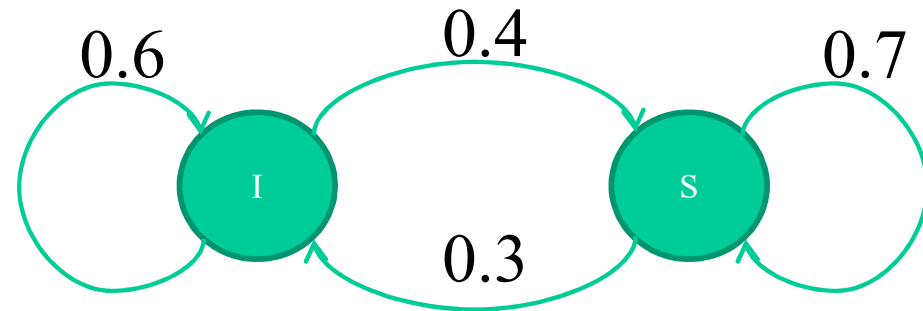
Neuron Firing



Transition Matrix

$$P = \begin{pmatrix} 0.6 & 0.3 \\ 0.4 & 0.7 \end{pmatrix}$$

Sum to 1 Sum to 1



- A Russian mathematician Andrei Markov (1856-1922) developed the theory to find out a *steady state vector* x so that $Px = x$.
- This is a perfect application for the eigenvector theory in **linear algebra** (線性代數).



Markov Chain

- Suppose I and S are the initial population of the **inner city** and the **suburban area**,
- Assume that every year 40% of the inner city population moves to the suburbs, while 30% of the suburb population moves to the inner part of the city
- After one year the populations are

$$\begin{pmatrix} 0.6I + 0.3S \\ 0.4I + 0.7S \end{pmatrix} = \underbrace{\begin{pmatrix} 0.6 & 0.3 \\ 0.4 & 0.7 \end{pmatrix}}_{\text{Transition Matrix}} \begin{pmatrix} I \\ S \end{pmatrix}.$$

最簡單的文字產生器

■ 訓練階段：先收集足夠多的同性質文章。

- 計算出相鄰兩字出現的頻率
- ‘學’: { ‘校’:2, ‘生’:2, ‘院’:1 }

■ 產生文字時，根據前一個字，用機率決定下一個字為何。

- 「學校」: 2/5, 「學生」: 2/5, 「學院」: 1/5

☞ 高雄海大學院本周將互補或整併建立共通科大學生，期待高雄海科大、第一科技大。行政院將可順利過到了最大學校。行政院將可順利過併後，包括工程、休閒觀光等各校的科大及高海科目學生數將可順利過併建立高應大學校長楊慶煜

Two-Dimensional Dictionary

■ 最愛去愛河坐愛之船

- {'最': {'愛': 1},
'愛': {'去': 1, '河': 1, '之': 1},
'去': {'愛': 1},
'河': {'坐': 1},
'坐': {'愛': 1},
'之': {'船': 1}}
}

Markov Text Generator

- With the statistics of 2-grams,
- For a given word, we have a list of words as candidates which may follow it.
- The probability for each following word is different, so we can model this with a random number.
 - A two-dimensional dictionary
 - {word_a: { word_b : 2, word_c : 1, word_d : 1 },
word_e : { word_b : 5, word_d : 2 }, ... }

buildWordDict()

- buildWordDict() builds a 2-dimensional dictionary
 - In this program, punctuations and newline (\n) are not removed. They are treated like a word.

```
def buildWordDict(fn):  
    # fn can be either a filename or a list of filenames  
    d = {}  
    if type(fn) == list:  
        for f in fn:  
            body = readPoemBody(f)  
            addWord(body, d)  
    else: # a single filename  
        body = readPoemBody(fn)  
        addWord(body, d)  
    return d
```

readPoemBody(fn)

```
infile = open(fn, "r")
# Skip the first 3 lines (title, author, separator)
title   = infile.readline()[:-1]
author  = infile.readline()[:-1]
separator = infile.readline()[:-1]
if separator != ":
    print("[Warning] 3rd line of {} not empty\n {}".format(
        fn, separator))
else:
    print("Reading {}({})".format(title, author))
body = infile.read()
infile.close()
return body
```

addWord()

```
def addWord(words, wordDict):  
    words = words.replace("?", "")
```

```
    for i in range(1, len(words)):  
        if words[i-1] not in wordDict:
```

```
            #Create a new dictionary for this word
```

```
            wordDict[words[i-1]] = {}
```

```
            if words[i] not in wordDict[words[i-1]]:
```

```
                wordDict[words[i-1]][words[i]] = 0
```

```
                wordDict[words[i-1]][words[i]] += 1
```

```
    return wordDict
```

words: 愛河左岸
i: 1
words[i-1]: 愛
words[i]: 河

wordDict["愛"] = {}

wordDict["愛"]["河"] = 0
wordDict["愛"] = {"河": 0}

wordDict["愛"] = {"河": 1}

main()

```
from random import choice
fn = ['gdrive/MyDrive/war01.txt', 'gdrive/MyDrive/war02.txt']
wordDict = buildWordDict(fn)

#Generate text of length 100
MAX_LEN = 100
initialWord = choice(list(wordDict.keys()))
text = ""
currentWord = initialWord
i = 0
while True:
    i += 1
    text += currentWord
    if currentWord not in wordDict:
        currentWord = initialWord
    currentWord = retrieveRandomWord(wordDict[currentWord])
    if currentWord == '\n' and i >= MAX_LEN: break

print(text)
```

‘學’: { ‘校’:2, ‘生’:2, ‘院’: 1 }

retrieveRandomWord()

```
def wordCountSum(wordCount):  
    sum = 0  
    for value in wordCount.values():  
        sum += value  
    return sum
```

{ '校':2, '生':2, '院':1 }

```
def retrieveRandomWord(wordCount):  
    from random import randint  
    randIndex = randint(1, wordCountSum(wordCount))  
    for word, value in wordCount.items():  
        randIndex = randIndex - value  
        if randIndex <= 0:  
            return word
```

‘校’, when randIndex = 1, 2
‘生’, when randIndex = 3, 4
‘院’, when randIndex = 5

Training Text: 余光中

《白雲》

- 溫柔的雪啊你什麼也不肯說
- 嚶嚶婉婉謎樣的叮嚀
- 向右耳，向左耳
- 那樣輕的手掌溫柔的雪啊
- 那樣小的唇

《江湖上》

- 一雙鞋，能踢幾條街？
- 一雙腳，能換幾次鞋？
- 一口氣，嚥得下幾座城？
- 一輩子，闖幾次紅燈？
- 答案啊答案
- 在茫茫的風裡裏



左岸詩人 PK 少女小冰



樣白的雪啊
那樣輕的手掌溫柔的嘴唇
且張嘴唇
一口氣，算不算少年輕幾歲
在愛斯基摩的風裏
你該叫白的手，惡夢在茫茫的風裏
向右耳，算永遠
答案啊答案啊你什麼樣的潔癖溫柔
的雪啊
一開就有那樣輕的風裏
答案啊你的國

假如这是人们的恋情
淙淙的溪水展示着生命的活力
这就是人们幸福的日子





Try to train your AI_Poet

Input:

輕輕地我走了
正如我輕輕地來
我輕輕地招手 做別西天的雲彩
那河畔的金柳 是夕陽中的新娘
波光裡的艷影 在我心頭蕩漾
軟泥上的青荇 油油地在水底招搖
在康河的柔波裡 我甘心做一條水草
那餘蔭下的一潭 不是清泉 是天上的虹
揉碎在浮藻間 沉澱彩虹似的夢
尋夢撐一隻長篙 向青草更青處漫溯
滿載一船星輝 在星輝斑斕裡放歌
但我不能放歌 悄悄是離別的笙蕭
夏蟲也為我沉默 沉默是今晚的康橋
悄悄地我走了
正如我悄悄地來
我揮一揮衣袖 不帶走一片雲彩
尋夢撐一隻長篙 向青草更青處漫溯
滿載一船星輝 在星輝斑斕裡放歌
但我不能放歌 悄悄是離別的笙蕭
夏蟲也為我沉默 沉默是今晚的康橋
悄悄地我走了
正如我悄悄地來
我揮一揮衣袖 不帶走一片雲彩

■ Output:

沉默是天的笙蕭
那餘蔭下的艷影 沉澱彩虹
尋夢撐一船星輝斑斕裡的新娘
我揮一潭 是清泉 在星輝斑斕裡 我
悄地我沉默是天的一片雲彩虹
夏蟲也為我悄悄是夕陽中的新娘
我輕輕輕輕地我悄地我不能放歌
但我悄是離別西天的青處漫溯

Optional: Avoid Pitfalls

if words[i-1:i+1] in ["小大", "七入"]:

避免學到一些無意義的詞

pass

else:

if words[i-1] == words[i]:

避免一個字連續出現太多次
(e.g., “問問問問問問問”)

wordDict[words[i-1]][words[i]] = 1

else:

wordDict[words[i-1]][words[i]] += 1

機率壓低，但不要是
0。這樣才可能寫出
「輕輕地我走了. 正
如我輕輕地來。」

輕輕地我走了 (只用四句)

■ Before

- 走了
- 正如我輕地我輕輕輕輕地來
- 我走了
- 我輕地我輕地招手
- 正如我輕地我輕輕地我輕輕地招手
- 正如我輕輕地招手
- 我輕輕輕輕輕輕地招手
- 做別西天的雲彩
- 做別西天的雲彩
- 正如我輕輕地我輕輕地招手
- 正如我輕輕地招手

重覆的字

■ After

- 走了
- 做別西天的雲彩
- 正如我輕地來
- 正如我輕地我走了
- 我走了
- 做別西天的雲彩
- 做別西天的雲彩
- 我走了
- 正如我走了
- 做別西天的雲彩
- 我輕地我輕地招手
- 正如我走了

訓練量少，造不出新句子。

Homework: 藏頭詩

高貴的搖籃鳳梨與月亮
雄踞在風裡
女友的絕句
中國夢也結冰

高於那頭
子昂的長城要撥出一盞燈
那扭曲變貌的滋味
背後還是鄉愁的峭壁
黃河是鄉愁
一驚醒來
那小亭子盡成英雄

有時句子看起來怪怪的

- 車票在**茫**的風裡裏
- 捍衛領土**主**與國人
- 將我們最近的**夏天**才每秒都和未來飛降臨風景

只要出現一個奇怪的詞，
整個詩意就被破壞了！

Jieba 中文斷詞

```
import jieba  
for s in ["台北市市民福利最好",  
         "台北市市民主程度不佳",  
         "提高高雄女學生全面的國文創意",  
         "高雄女學生最喜愛的文創產業",  
         ]:  
    tokens = jieba.lcut(s)  
    print(s, "->", "/" .join(tokens) )
```

它如何斷詞？

1. 統計
2. 文法


台北市市民福利最好 -> 台北/市民/福利/最好
台北市市民主程度不佳 -> 台北市/民主/程度/不佳
提高高雄女學生的國文創意 -> 提高/高雄/女/學生/的/國文/創意
高雄女學生最喜愛的文創產業 -> 高雄/女/學生/最/喜愛/的/文創/產業

addWord() 只需加兩行

```
def addWord(words, wordDict):  
    import jieba  
    words = words.replace("?", "")  
    words = jieba.lcut(words)  
  
    for i in range(1, len(words)):  
        if words[i-1] not in wordDict:  
            #Create a new dictionary for this word  
            wordDict[words[i-1]] = {}  
        if words[i] not in wordDict[words[i-1]]:  
            wordDict[words[i-1]][words[i]] = 0  
  
            wordDict[words[i-1]][words[i]] += 1  
  
    return wordDict
```

加上斷詞功能後

- 整個太平洋洶湧的，
- 默默的面容，一叢叢的孤兒
- 但是只發現他背影。
- 刺眼地閃出了地中海岸邊金黃色的甘釀。
- 我而遺下的尾巴，
- 縛不住時間的漁夫
- 還嗅到你是禿子，是那虹，
- 當寂寞來襲時，時來俯窺自己的丁香
- 我的小屋。



至少比較沒有奇怪的詞出現了

Test with 再別康橋

更青處漫溯
夏蟲也為我走了
但我不能放歌 沉默 悄悄是離別的虹
我揮一揮衣袖 向青草更青處漫溯
滿載一船星輝 斑斕裡放歌
尋夢撐一隻長篙 向青草更青處漫溯
滿載一船星輝 悄悄是今晚的青荇 是離別的新娘
悄悄地來
揉碎在水底招搖
但我悄悄地來
尋夢撐一隻長篙 不帶走一片雲彩
悄悄地我不能放歌
揉碎在浮藻間 不是清泉 是夕陽中的新娘

繁體中文語料庫 (corpus)

```
from google.colab import drive  
drive.mount('/content/gdrive')
```

```
import os  
os.chdir('/content/gdrive/MyDrive')  
os.mkdir('dict')
```

```
from urllib.request import urlretrieve  
urlretrieve('http://poet.ncnu.org/dict/dict.txt.big', 'dict/dict.txt.big')
```

jieba.lcut()

- `import jieba`
- `jieba.initialize('dict/dict.txt.big')`
- `s = '輔導處懇請同學面試後撥空協助，將你寶貴的甄選經驗及備審資料(學檔等)提供給未來的學妹們參考，相信學姊們的傳承與支持，會是學妹們在迷茫中非常重要的助力與指引'`
- `print('/' .join(jieba.lcut(s)))`
 - ☞ 輔導處/懇請/同學/面試/後/撥空/協助/，/將你/寶貴/的/甄選/經驗/及備/審/資料/(學檔/等)/提供/給/未來/的/學妹/們/參考/，/相信/學姊/們的/傳承/與/支持/，/會/是/學妹/們/在/迷茫/中/非常/重要/的/助力/與/指引

add_word()

■ jieba.add_word('備審', 1, 'an') # 名形詞

- word, frequency, tag

■ print('/' .join(jieba.lcut(s)))

☞ 輔導處/懇請/同學/面試/後/撥空/協助/，/將你/寶貴/的/甄選/經驗/及/備審
/資料/(/學檔/等)/提供/給/未來/的/學妹/們/參考/，/相信/學姊/們的/傳承/
與/支持/，/會/是/學妹/們/在/迷茫/中/非常/重要/的/助力/與/指引

詞性列表

■ <https://gist.github.com/luw2007/6016931>

- v: 動詞
- n: 名詞
 - ☞ nr: 人名 (ren = 人)
 - ☞ ns: 地名 (space = 空間)
 - ☞ nz: 其他專有名詞 (e.g., 全真教) (zhuan = 專)
- a: 形容詞 (adjective)
- p: 介系詞 (preposition)
- d: 副詞 (adverb, 但a被形容詞用掉了)
- uj: u 助詞 (auxiliary 的第二個字母)
- x: 字符串 (標點符號)

詞性 (Part of Speech) 判斷

```
import jieba.posseg
```

```
sentence = '天下風雲出我輩，一入江湖歲月催。'
```

```
aList = jieba.posseg.lcut(sentence)
```

```
for word, tag in aList:
```

```
    print(word, tag, sep='\t')
```

天下	s	處所詞 (space), 如「山上」「海底」
風雲	n	名詞 (noun)
出	v	動詞 (verb)
我輩	r	代名詞 (pronoun), 如「你我他們」
,	x	標點符號 (punctuation)
一	m	數量詞, 如「五」「千」
入	v	
江湖	ns	地名 (n space), 如「巴黎台北」
歲月	m	量詞, 如「匹」「輛」
催	v	
。	x	

若需調整，同樣可以
`jieba.add_word('高雄女中', 1, 'nt')`

七言詩上四下三

```
poem = "黃河遠上白雲間
```

```
一片孤城萬仞山
```

```
羌笛何須怨楊柳
```

```
春風不度玉門關"
```

```
for s in poem.split():
```

```
    print( '/' .join( jieba.lcut(s) ) )
```

- 黃河/遠上/白雲/間
- 一片/孤城/萬仞/山
- 羌笛/何須/怨/楊柳
- 春風/不度/玉門關

Homework: `load_userdict()`

- Select 10 new poems and let Jieba segment them.
- Discuss with your team members which phrases you think Jieba did not segment correctly.
- Suppose you have many rules you want to teach Jieba, writing them one by one with `add_word()` will be troublesome.
- You may write them in a text file and `jieba.load_userdict('mydict')`
 - Each line in 'mydict' consists of 3 fields: word, frequency, tag.
 - Separated by a space.

Activity: 典故與推敲

換成「靜」會
不會更好？

- 是供品默默 紅似我追悔
- 是檀香冉冉 悠似他歸家

語出何處？

冉冉檀香透過窗 心事我了然
~ 周杰倫《青花瓷》

Activity: 唐詩三百首 (or 余光中600首)

全班30人，每人分到10首。

- 回去練習把這10首用Jieba進行斷詩。小組互相檢查有沒有不自然，必須例外處理的詞。
- 練習標註這10首詩中，哪幾首有「春夏秋冬」、「愛恨情愁」、「鳥獸花草」、顏色、溫度、聲音。
 - 銀燭秋光冷畫屏，輕羅小扇撲流螢。
 - 天階夜色涼如水，臥看牽牛織女星。
- 小組相互討論標註是否合宜。有爭議的詩句，提出到課堂上，請老師講解。
 - ☞ 學生自學、組內共學、組間互學、教師導學
- 藉由AI機器人的協助，每週練習寫10個含有唐詩元素的句子。
 - 每週一個主題：愛情、季節、顏色、溫度、聲音.....

意象