

Rime 打嘸蝦米

嘗試

- [myliu](#)：似乎是從官方 GCIN 碼表轉來的
 - [【介紹】使用 Rime 在各大大平臺安裝嘸蝦米輸入法 | Yu-Cheng Huang](#)
 - 看來沒有簡拼、記憶詞組等功能，就是模擬官蝦的組字上字而已
- [ianzhuo/rime-liur](#) (PTT [afrink](#) 的「蝦拼」) (介紹續篇)：嘸蝦米與拼音混輸，似乎支援簡拼、組詞
 - 但不太知道怎麼安裝及自訂，且功能複雜很多 @@ (作者厲害啊)
 - ²⁰¹⁹⁰³²⁰ 今天總算有進展：
 1. 寄 PTT 站內信給作者，得知手動安裝的話，/opencc 也要複製到 Rime 的資料夾 我留著 default.custom.yaml 及 weasel.custom.yaml 兩個，其他都複製過去了。然後就是研究看看他的 default.custom.yaml 裡哪些要整合到我目前的去.....
 2. 先不管 default.custom.yaml 只單純加入 liur 這個 schema 終於開始試用了，現在就是用這個「蝦米輸入法」來打的，速度也蠻快的不會卡頓
 1. 原來不支援簡拼，可惜。我想要的「簡拼嘸蝦米」可能得參考 Rime 官方的倉頡方案了
 2. 候選字清單中除了目前可組的字外，也會提示再打什麼碼可以組出其他的字，這應該是 Rime 本身的能力吧？太強大了
 3. 打標點符號遇到問題，`` 無法正常出來
 4. ``Caps Lock`` 及 `` 都可用來切換英/漢，不錯
 5. 連續的大寫字母的話是按空白鍵出來
 6. ```` 鍵叫出拼音模式
 7. 第一次使用 Rime 的反查模式，挺不錯的
 8. 簡體漢字、日體漢字、擴充漢字也整合在一起幾乎都可以直接打
 3. **【結論】** 這個「蝦米輸入法」相當程度可替代官蝦
 - 問題
 - **FIXED** s. ≠ □
.. ≠ □ 把 key_bindings:/paging_with_comma_period 取消掉就解決了¹⁾
 - 數字選序都會比官蝦偏離 1 個 (以前用 LIME 也是這樣)
- 在 Mac 下使用鼠鬚管(RIME) + 嘸蝦米，並擁有注音模式、讀音反查、中英混輸、動態造詞 | 只是個打字的 (以 afrink 版為基礎的改良) 待測試 
 - 照他的修正，動態造詞可以用了~
- ²⁰¹⁹¹⁰²¹ [RESTART](#) 並記錄在本篇及 [鼠鬚管 Rime](#) 中 (過程中先回來用官蝦，已經有點不習慣了，主要是差在切換漢/拉是用 Caps 還是 ⇧，以及要輸入拉丁字母時可否直接開始打字)
- ²⁰¹⁹¹¹⁰⁸ 最近有感於 liur 一些特殊符號的碼序和官蝦不太一樣 (想發 issue 耶.....)，而 afrink 又繼續做了一個 liur-lua 也有一些新功能可以整合研究，所以可能最近會再試試不同的方案
 - 之前的 myliu 特殊符號碼序如何？ 更慘，缺很多特殊符號耶 myliu.dict 檔案大小 313 KB liur_Trad.dict 就有 341 KB
 - 另外，剛好前兩天教會弟兄也回饋給我一份前人做的 CIN 檔，也許研究看看那份詞庫
 - 還有也許我也自己再從官網去下載工具或自己轉檔 丟給 chitsaou 的 rime-boshiamy 結果還不錯，碼序看來都是對的，和官蝦一樣，只是少了 liur 的很多功能，不過漢拉混打、`` `` 切換漢拉是有的。真好奇為什麼 liur 會有碼序跑掉的問題？

方案

- [chitsaou/rime-boshiamy](#) 可能是最舊的
 - 但當年也有一些不錯的研發及嘗試，像試著做連打/簡拼或注音反查，有兩個 branch 可參考：[Branches · chitsaou/rime-boshiamy](#)
- [ianzhuo/rime-liur](#)

- [hsuanyi-chou/ rime-liur](#) fork 及修改
- [ianzhuo/ rime -liur-lua](#) 好像更新！
 - 又有高手再 fork 這個，可自動安裝，並且大幅整理過！[hftsai256/rime-liur-lua: lua擴充版^{2\)}](#)
- [amoshyc/ rime - liu](#) (較舊，最後更新是 2016)
 - 同作者的 [amoshyc/myliu](#) 較新，最後更新是 2018³⁾
 - 新舊版都下載來比較了一下，碼表一模一樣，只有 `schema.yaml` 稍有增補而已

安裝

根據先前嘗試的經驗，整理如下：

1. 先跑完鼠鬚管的安裝

1. 下載及安裝 | RIME | 中州韻輸入法引擎 | macOS 鼠鬚管

- PKG 法 (略)
- ²⁰¹⁹¹⁰²¹ 之前已試過 PKG 安裝法，今天也來嘗試 `brew cask install squirrel`，因為我之前有裝過 Homebrew
 - 1. 花了一小段時間更新 Homebrew 後順利裝上，似乎需要重新登入系統
 - 2. 重新登入系統回來，發現系統輸入法清單中已有鼠鬚管，不確定是不是 Homebrew 弄的。另外 `~/Library` 下目前還沒有 Rime 資料夾
 - 3. 有了，第一次切到鼠鬚管後，就會自動出現 Rime 資料夾了。然後預設的輸入方案是剛月拼音。這樣就裝好了。如果只要打拼音或注音的話，應該到此為止

2. 東風破裝注音 | `curl -fsSL https://git.io/rime-install | bash`

- 其實這樣一次預裝的是好幾個輸入法
- 咦，其實我忘記確認剛才用 Homebrew 裝好鼠鬚管後是否就已經有注音了。如果有的話，其實以後也許不用再跑東風破這一步了
稍晚證實，較新版的鼠鬚管已不再需要另用東風破裝注音

3. 自行裝入其他輸入方案

- 嘸蝦米

2. 下載 [hsuanyi-chou](#) 修正過的 `rime-liur`，然後把 `/Rime` 裡的 YAML 檔都放到 `~/Library/Rime`

- 若照 [afrink](#) 的說法 `~/openccc` 也要放 = 全放了⁴⁾

3. 重新部署鼠鬚管就可以使用了

功能

{ 以 liur 為主 }

- **自動造詞** 後來的 liur 似乎已經沒有這個功能了
- **手動造詞** (用每字的首碼組成) : ``;`` 進入此模式
 - `;nuj`wuo`ae`h+` ` ` ` → 宇冶金時` 之後可打 `uwah+` ` ` `` 出此詞
 - 若不小心造錯了詞或想刪某詞：打出該詞但不要按 `` 送出，而是按 `` 即可刪去
- **自行加詞** (更自由，相當於官蝦的「加字加詞») :
 - 我試過直接放 `liur.extended.dict.yaml` 應該就行了，但 [hsuanyi-chou](#) 說要放在 `liur_customWords.dict.yaml` 裡；anyway 我還是先放前者裡，有「人工編碼詞」段落
 - 注意這個檔案中，詞組之間要用 `tab` 但其他 YAML 部分的縮排 `indent` 還是要用兩個兩個空格 `Sublime Text` 之類的編輯器可以自動轉換 `indent` 如果不小心把 YAML 的空格縮進轉換成 `tab` 部署 Rime 時就會出錯！要小心！

調整

- 若下載的是 [ianzhuo](#) 的 [原版 rime-liur](#)，則可先照 [hsuanyi-chou](#) 的 [歷次 commit](#) 逐一手動改入

剛裝好 hsuanyi-chou 版的狀態	調整方法
組碼沒有顯示在輸入區	squirrel.custom.yaml 的 style/inline_preedit 改為 true
mac_white2 主題..... 我不行	改 squirrel.custom.yaml 的 style/color_scheme，可先選回預設的，或自製一個 <方案名>_theme.yaml 後指定使用
Times New Roman 字型..... 我不行	改 squirrel.custom.yaml 的 style/font_face
其他各項候選框樣式設定	改 squirrel.custom.yaml 的 style/xxxxxxx (有機會再詳述) ⁵⁾
選字號碼與官蝦差 1	在 liur.custom.yaml 中加入 <pre> menu: alternative_select_keys: "0123456789" </pre>
啟用日文漢字	在 liur.extended.dict.yaml 中把 # - liur_Japan 這行取消註解
候選視窗每頁項目數	default.custom.yaml ⁶⁾ 中設定 menu/page_size，我都設 9

剛裝好 hsuanyi-chou 版的狀態	調整方法
各項熱鍵	<pre>default.custom.yaml[] 想直接用： key_binder: bindings: ... __patch: - key_bindings:/paging_with_minus_equal # =、 - 翻頁 - key_bindings:/numbered_mode_switch # Ctrl+Shift+ 數字鍵切換模式 但失敗了。似乎得直接把 /Library/Input Methods/Squirrel.app/Contents/SharedSupport/key_bindings.yaml 裡的設定複 製出來，變成： key_binder: bindings: ... #paging_with_minus_equal: - { when: has_menu, accept: minus, send: Page_Up } - { when: has_menu, accept: equal, send: Page_Down } #numbered_mode_switch: - { when: always, accept: Control+Shift+1, select: .next } - { when: always, accept: Control+Shift+2, toggle: ascii_mode } - { when: always, accept: Control+Shift+3, toggle: full_shape } - { when: always, accept: Control+Shift+4, toggle: simplification } - { when: always, accept: Control+Shift+5, toggle: extended_charset } - { when: always, accept: Control+Shift+exclam, select: .next } - { when: always, accept: Control+Shift+at, toggle: ascii_mode } - { when: always, accept: Control+Shift+numbersign, toggle: full_shape } - { when: always, accept: Control+Shift+dollar, toggle: simplification } - { when: always, accept: Control+Shift+percent, toggle: extended_charset }</pre>
摺疊選單選項	
將大寫字母排除在組字碼外，大寫字母直接上字	
用 `=``-`` 翻頁	同上「各項熱鍵」

20191105 今天看到 hsuanyi-chou 說他更新了 liur 一看原來是採納了我把 1234567890 改為 0123456789 仿官蝦的修改；還有字型從 Times New Roman 改為更好看的 (剛好我也愛用 Operator Mono 也太巧!)。原來我的資訊文還是能被搜尋得到，還有人看啊！

自訂

- 某些軟體內改為拉丁字母優先，或優先打出半形標點，或預設打出簡體字[]xxxxxxx.custom.yaml 內設定

```
app_options:
  "com.runningwithcrayons.Alfred-3":
    ascii_mode: true
    ascii_punct: true      # 這是我自己試出來的！
    simplification: true  # 一般線上文件沒有提到除了 ascii_mode 之外的設定！
```

- Welly[]org.net9.Welly
- 自製擴充的 table
 1. 弄 <表格名>.dict.yaml
 2. 在 liur.extended.dict.yaml 中用 import_tables 引用

碼表

1. 買官蝦 自己取出轉檔
2. 現成
3. 貢獻 [開放格式蝦米編碼庫\(openxiami\)](#)

問題

- 反查讀音目前無效 用 ; ; 開啟
- translator/enable_completion 目前關掉也沒用？
後來可以了，fixed/enable_completion 也 false 就行了
- liur 的一些特殊符號碼序似乎會隨著使用過程而變得與官蝦不同？
 - ²⁰¹⁹¹¹¹⁰ 把現用的 liur 所有檔案備份起來，重新下載最新版的 hsuanyi-chou liur
 - 字序果然和官蝦是一樣的！
 - 也是會自動產生 liur.txt[]而且內容和之前的一模一樣，真神奇。不過目前好像不影響詞序。不知道常選我要的詞以後會不會改變。
 - ²⁰¹⁹¹¹¹⁷ 今天有新發現 要修正之前的
 1. 字序會不會變，關係在於 liur.extended.dict 內有沒有開啟 liur_Japan[]開啟的話字序就不對了
 - afrink 說可能要用 openxiami[]我試了一下，新的 openxiami_TCJP.dict 字序比較正確
 2. 在重新部署前，要把 /build 也刪掉，才會重新產生 liur.txt (是否產生 liur.txt 也不重要，重點是字序應該是被存到 /build 的二進位檔中，不知道是 .prism 還是哪個)

1)

default.custom.yaml 中，加入：

```
key_binder:
  bindings:
    __patch:
      - key_bindings:/emacs_editing
      - key_bindings:/move_by_word_with_tab
      - key_bindings:/paging_with_minus_equal
#      - key_bindings:/paging_with_comma_period
      - key_bindings:/numbered_mode_switch
```

也許還有更好的方法

2)

Windows 上似乎已無法自動安裝，所以我整理了一些步驟，詳見

<https://www.ptt.cc/bbs/Liu/M.1585969156.A.97A.html>

3)

有 7 個 fork 但都沒有新 commit 推測是大家不小心按到又不知道怎麼刪專案？

4)

檔案清單

```

default.custom.yaml
liur_customWords.dict.yaml
liur_Japan.dict.yaml
liur_Trad.dict.yaml
liur_TradExt.dict.yaml
liur_TradToSimp.dict.yaml
liur.custom.yaml
liur.extended.dict.yaml
liur.schema.yaml
squirrel.custom.yaml
weasel.custom.yaml
opencc/liu_w2c.json
opencc/liu_w2c.txt
opencc/liu_w2cExt.txt
opencc/liu_w2cJp.txt

```

5)

```

# style/font_face: Alegreya Bold, Songti TC Bold # 我喜歡的字體名稱
# style/font_face: Alegreya Bold, Typeland KhangXi Dict Trial # 我喜歡的字體名稱
style/font_face: "Operator Mono Bold, HanziPen TC Bold, HanziPen SC Bold"
# 我喜歡的字體名稱
style/font_point: 20 # 字號
style/corner_radius: 5 # 窗口圓角半徑
style/border_height: 7 # 窗口邊界高度，大於圓角半徑才有效果
style/border_width: 7 # 窗口邊界寬度，大於圓角半徑才有效果
style/line_spacing: 0
# style/spacing: 0
style/label_font_face: DFLiKingHei-XB #Operator Mono Bold Italic # 編號的字型
style/label_font_point: 12
style/hilited_corner_radius: 5 # 高亮度區的圓角
style/candidate_format: "%c %@"

```

6)

其實 liur 的這個檔是用 Weasel 產生的

From: <http://ciangbrides.myds.me/dw/> - **Ciang Brides** 黃金城

Permanent link: <http://ciangbrides.myds.me/dw/it/rime/liu>



Last update: **2021/07/16 18:27**

