自製中文選字系統：基礎篇

課程問題 | 本日作業 | 問題討論 | 學習心得(完成)

重要知識點

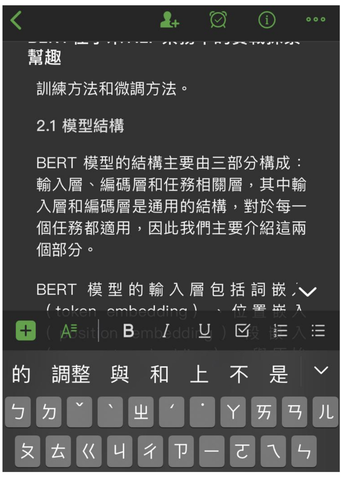
- 運用以下所學來完成此專題：Python String 基礎操作、斷詞、基礎語言模型
- 學習如何用 Ngram 完成一版簡易的中文選字系統

實作提示

請搭配 Jupyter Notebook 使用本教材

中文選字系統

- 本系列是實務專題，我們將讓大家著手完成一個「 中文選字系統 」
- 你將會運用以下所學來完成此專題：Python String 基礎操作、斷詞、基礎語言模型



一個選字系統要具備什麼功能？

- 根據上文推薦接續的文字
- 推薦的接續文字可長可短，通常介於 1 到 4 個字
- 比較常出現的字詞應該要往前排列

資料源：中文Wiki

- 格式：“<標題>\t<內文>” 共 119 篇
- 範例：英語 英語英語（，）又稱爲英文，是一種西日耳曼語言，誕生於中世紀早期的英格蘭，如今爲有全球通用語的地位，“英語”一詞源於邁居英格蘭的日耳曼部落盎格魯（），而“盎格魯”得名於脂波羅的海的半島盎格里亞（）。弗裏西語是與英語 ...
- 我們將用這個資料源來建立我們的選字系統

資料前處理

- 我們想要利用這個資料源來建立語言模型
- 仔細觀察數據，你會發現有許多非中文的文字在其中，包括：英文、其他外語、數字...等等
- 除了這些非中文字以外，我們也不希望標點符號出現在我的選字系統當中，所以這些文字都要去除掉
- 當我們去除這些文字後，因為語意已經被中斷了，所以在處理上我們需要將它斷開
- 這一排除非中文字比較難做，採取正面表列的方式比較好處理，我們只需要抓出「中文字元」即可
- 表示中文字元的 Unicode 為：
  - \u4E00-\u9FFF（常見）
  - ...（其他罕用字元詳見 wiki ）
- 使用 re.findall(...) 找出連續一段的中文字詞

使用 Ngram 來建立第一版選字系統

用之前介紹的方法建立 Ngram

Unigram

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i)$$

Bigram

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i|w_{i-1})$$

Trigram

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i|w_{i-2}w_{i-1})$$

當前面**沒有字**時，使用 Unigram：

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i)$$

當前面**有一個字**時，使用 Bigram：

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i|w_{i-1})$$

當前面**有兩個字**時，使用 Trigram：

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i|w_{i-2}w_{i-1})$$

下一步：完成作業