自然語言處理第一次作業

Q1：Which embedding model do you use? What are the pre-processing steps? What are the hyperparameter settings ?

在TODO2中，我使用Gensim中的 Google 預訓練 Word2Vec 模型 (GoogleNews-vectors-negative300.bin)

在TODO5中使用自訓練的Word2Vec模型，以下為訓練前處理(Pre-processing)的方式：

- 使用 gensim.utils.simple_preprocess 進行 tokenization
- 使用 gensim 的內建停用詞表 (STOPWORDS) 移除停用詞

```
# 以下程式碼由我個人整理Claude 3.7 和 ChatGPT o3-mini-high，所給予的不同使用方式
# pre-process(Claude)
# 移除非英文字符 (`remove_non_english`)
def remove_non_english(text):
    return re.sub(r'[^a-zA-Z\s]', ' ', text)
```

- 使用正則表達式：^a-zA-Z\s，他會識別且替換掉"非"英文和"非"空格字符

```
#詞形還原器
def basic_lemmatize(word):
```

•處理常見的英文詞尾變化，如：

- 複數形式：`cats` → `cat`
- 進行時：`running` → `run`
- 過去式：`played` → `play`

```
#文檔預處理 (`preprocess_document`)
def preprocess_document(text, min_word_length=3, max_word_length=15):
```

- 這是核心預處理函數，整合了所有處理步驟

```
# 語料庫處理 (`process_corpus`)
def process_corpus(input_file, min_word_length=3, max_word_length=15, min_word_freq=5):
```

- 這個函數處理整個語料庫文件

```
#來自ChatGPT的預訓練處理

```

- 模型中我使用的超參數為以下python程式碼：

#TODO 5

```
model = Word2Vec(
    cores = multiprocessing.cpu_count(),
    sentences = corpus,
    vector_size = 200, # 詞向量維度 (100-300之間)
    window = 5, # 上下文窗口大小
    min_count = 5, # 最小詞頻
    epochs = 10, # 訓練迭代次數
    sg = 1
)
```

- 後來，我因為訓練太多次且結果都很差，於是我有調整一些超參數

#TODO 5 another version

```
model = Word2Vec(
    cores = multiprocessing.cpu_count(),
    sentences = corpus,
    vector_size = 200, # 詞向量維度 (100-300之間)
    window = 5, # 上下文窗口大小
    min_count = 2, # 最小詞頻
    epochs = 5, # 訓練迭代次數
    sg = 1
```

```
9   )
```

Q2：What is the performance for different categories or sub-categories?

```
Category: capital-common-countries
  Total analogies: 506
  OOV analogies: 506 (100.00%)

Category: capital-world
  Total analogies: 4524
  OOV analogies: 4524 (100.00%)

Category: currency
  Total analogies: 866
  OOV analogies: 866 (100.00%)

Category: city-in-state
  Total analogies: 2467
  OOV analogies: 2467 (100.00%)

Category: family
  Total analogies: 506
  OOV analogies: 86 (17.00%)
  Evaluated analogies (in-vocabulary): 420
  Correct predictions: 385 (91.67%)

Category: gram1-adjective-to-adverb
  Total analogies: 992
  OOV analogies: 122 (12.30%)
  Evaluated analogies (in-vocabulary): 870
  Correct predictions: 297 (34.14%)

Category: gram2-opposite
  Total analogies: 812
  OOV analogies: 0 (0.00%)
  Evaluated analogies (in-vocabulary): 812
  Correct predictions: 180 (22.17%)

Category: gram3-comparative
  Total analogies: 1332
  OOV analogies: 0 (0.00%)
  Evaluated analogies (in-vocabulary): 1332
  Correct predictions: 1067 (80.11%)

Category: gram4-superlative
  Total analogies: 1122
  OOV analogies: 0 (0.00%)
  Evaluated analogies (in-vocabulary): 1122
  Correct predictions: 512 (45.63%)

Category: gram5-present-participle
  Total analogies: 1056
  OOV analogies: 300 (28.41%)
  Evaluated analogies (in-vocabulary): 756
  Correct predictions: 421 (55.69%)

Category: gram6-nationality-adjective
  Total analogies: 1599
  OOV analogies: 1599 (100.00%)

Category: gram7-past-tense
  Total analogies: 1560
  OOV analogies: 0 (0.00%)
  Evaluated analogies (in-vocabulary): 1560
  Correct predictions: 983 (63.01%)

Category: gram8-plural
  Total analogies: 1332
  OOV analogies: 72 (5.41%)
  Evaluated analogies (in-vocabulary): 1260
  Correct predictions: 967 (76.75%)

Category: gram9-plural-verbs
  Total analogies: 870
  OOV analogies: 270 (31.03%)
  Evaluated analogies (in-vocabulary): 600
  Correct predictions: 333 (55.50%)
```

- 在使用 Google Word Analogy 資料集進行評估時，不同子類別的表現存在明顯差異。

Q3. What do you believe is the primary factor causing the accuracy differences for your approach? (5%)

我認為造成類比任務中不同子類別表現差異的主要原因是：**詞彙表覆蓋率（Vocabulary Coverage, OOV）差異**以及**語料中不同類別詞彙的出現頻率與語言結構特性。**

OOV 比例主導了能否進行預測：以下為利用第二題所整理之佐證：

從類比任務的輸出結果可見，部分子類別完全無法進行預測，原因是：

子類別	OOV 比例
capital-common-countries	**100.00%**
capital-world	**100.00%**
currency	**100.00%**
city-in-state	**100.00%**
gram6-nationality-adjective	**100.00%**

這些類別題目雖然很多（像 capital–world 有 4524 題），但因所含詞彙如地名、幣別、國籍形容詞等在訓練語料中頻率低，導致通通被 min_count=5 過濾掉，或根本未出現於語料中，形成 **詞彙表外（OOV）問題**，完全無法參與評估。

語料頻率分布與語言結構影響準確率高低

對於 OOV 較低、可被有效評估的子類別，我觀察到準確率明顯不同，這與類別中詞彙的語言結構或語意規律有關：

類別	OOV 比例	正確率	說明
family	17.0%	**91.67%**	常見詞彙（如 king/queen/father）語意清楚
gram3-comparative	0%	**80.11%**	比較級規則一致（如 good → better）
gram8-plural	5.41%	**76.75%**	規則轉換清晰（如 cat → cats）
gram4-superlative	0%	45.63%	雖規則但較易混淆（如 best, greatest）
gram2-opposite	0%	22.17%	抽象語意關係，較難學習
gram5-present-participle	28.41%	55.69%	雖有 OOV，但規則可學

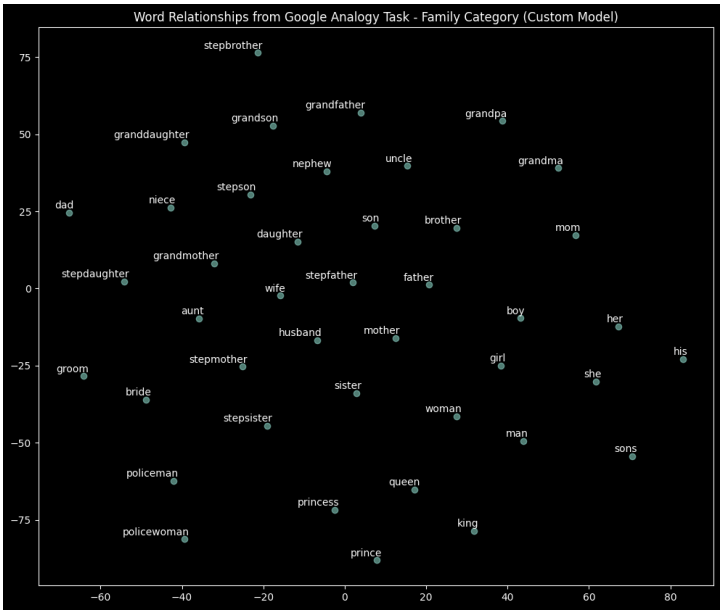
這顯示：

- **語言規則明確的子類別（如比較級、複數變化）模型表現較佳。**
- **語意類別（如相反詞、國籍）或結構複雜的則準確率偏低**，即使詞彙在詞表中，也未必能精準預測。

為改善準確率，之後訓練時可以考慮一下措施：

- 擴充語料（特別是含地名與專有詞彙）
- 降低 min_count
- 分類訓練詞向量（domain-specific embedding）

Q4：What's your discovery from your t-SNE visualization plots?



From 第四次模型訓練： 透過 t–SNE 視覺化圖發現，模型能捕捉某些清晰的語義群聚，例如性別或家庭關係的對應詞彙。然而部分詞彙的分佈仍較分散，顯示模型尚未完全掌握更細緻的詞彙語義結構。

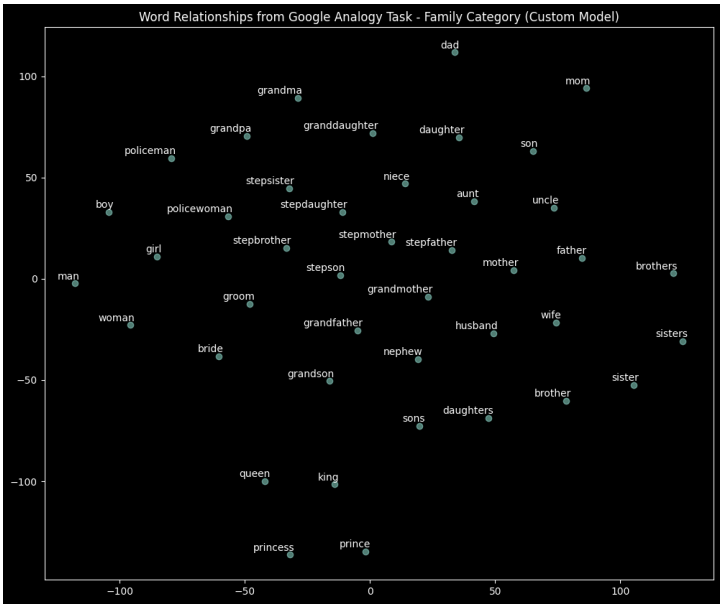
▽在第四次output中：

```
成功載入自訓練詞向量模型，詞彙量: 947422

開始進行詞彙類比預測...

100%|██████████| 19544/19544 [02:38<00:00, 123.61it/s] 完成預測！總共 19544 個類比問題中，有 14531 個問題包含詞彙表外的詞(74.35%)。

```



From 第五次模型訓練： 透過 t–SNE 視覺化圖發現，模型比上一張圖結構性更佳，詞義相似的詞終於沒有偏離至奇怪的位置了，而我只是將 min_count 降低至2，且epoch降低至5，表示再多一點遞迴次數，就可以使模型有更好的預測。

▽但在第五次output中：

```
成功載入wiki_word2vec.kv，詞彙量: 2081725
開始進行詞彙類比預測...
100%|██████████| 19544/19544 [10:19<00:00, 31.53it/s]  完成預測！總共 19544 個類比問題中，有 10812 個問題包含詞彙表外的詞(55.32%)。
```

Q5：What's the difference in word representations if you increase the amount of training data?

• 在這次實驗中，我將 Word2Vec 模型的訓練資料由原本的維基百科部分樣本擴充至更完整的語料，最終保留的詞彙量達 **2,081,725 個詞**，比先前模型的 94 萬詞大幅提升。

• 擴充資料後重新進行 Google Word Analogy 詞彙類比任務，總共 19,544 題中，**OOV 題目降至 55.32%**（即 10,812 題包含至少一個詞在詞彙表外），相較先前超過 74% 的 OOV，明顯改善。

差異：

項目	原始模型（小語料）	擴充語料後
詞彙量	約 947,422	2,081,725
OOV 題數	約 14,531 題	10,812 題
OOV 比例	74.35%	55.32%
可被評估的類比題數	約 5,000 題	約 8,732 題
類比任務表現（提升潛力）	低（資料稀疏）	中等（詞彙覆蓋較廣）

• **隨著訓練資料增加**，模型能學習到更多詞彙的上下文，尤其是過去被過濾的低頻詞，現在可以被納入詞彙表中，這不僅**提升了詞向量的語意豐富度**，也讓更多類比問題能夠被有效評估。

• 對於具專有名詞、地名、時態變化等的子類別，如 capital–common–countries、gram5–present–participle 等，這類擴充詞彙最有助益。

> 增加訓練語料的數量與多樣性，**可顯著提升模型的詞彙覆蓋率與語意學習能力**，特別是在詞彙類比任務中可以降低 OOV 比例，增加可被評估的題目數，進而提升整體表現與語意解析準確性。

Q6(Bonus)：Anything that can strengthen your report.

我訓練了大概五次以上，為了讓圖變得詞與詞之間的詞義相關性提高，我多做了其他像是預測完後的回答率，前幾次都是在OOV上的程度大約50%，後來第四次訓練時OOV達到更高，但這樣表示我的語料庫不夠或是訓練預處理沒有足夠好。

Enviroments	Local
Running Environment	Windows 11
Python version	Python 3.11.9