NLP 447 Project 2 Summary

Yuan Wang / ywang340

According to the instruction, there are four kinds of measurement methods implemented in this project:

* **the cosine similarity for word2vec vectors**

Use pre-trained model “﻿glove-wiki-gigaword-100” from ﻿genism package, where 100 indicates the number of dimensions between word vectors.

* **Wu-Palmer similarity on the TRIPS ontology**

Load the lex-ont.json file to get lf\_parents word. It is possible to have more than one lf-parents, so collect all the previous words in a list, then find the intersection of two lists as LCS. Choose the LCS with higher depth and the parents include the LCS, finally compute the similarity.

* **the cosine similarity for vectors computed using the Brown corpus**

Build N by N metrics based on the length of words number from ﻿trips-brown\_NV\_overlap.txt, update the metrics based on Brown corpus, then calculate the cosine similarity from vector of input words

* **Forth novel method**

Use the word2vec based on Brown corpus to predict similarity score

The output odd word is based on the similarity score, for each group (three words), if two of words get max score, the odd word is the rest of word

**Discussion:**

I labeled the assumed correct odd words for each group; it looks that the first and the second method have higher accuracy than others. However, my novel 4th method may correct for the last group of word which depends on how you interpretate the words. Waste and save can be opposite words, so help is the odd. The third method depends on the training data and looks the more the better.

**Output:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **tripleid** | **word2vec\_score\_choice** | **Wu\_Palmer\_score\_choice** | **brown-vector\_score\_choice** | **4th\_novel\_score\_choice** |
| **0** | house | house | cat | house |
| **1** | bottle | bottle | bottle | bottle |
| **2** | drum | health | drum | health |
| **3** | doc | friend | doc | friend |
| **4** | waste | waste | waste | help |

**Score:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **tripleid** | **word1** | **word2** | **word2vec\_score** | **Wu\_Palmer\_score** | **brown-vector\_score** | **4th\_novel\_score** |
| **0** | dog | cat | 0.8798075 | 1.0 | 0.8139726301062110 | 0.92693484 |
| **0** | dog | house | 0.43759328 | 0.36363636363636400 | 0.88328404000978 | 0.81000423 |
| **0** | cat | house | 0.37882093 | 0.36363636363636400 | 0.7008910505999740 | 0.5469933 |
| **1** | bottle | house | 0.27423105 | 0.42105263157894700 | 0.8158530834541380 | 0.8384596 |
| **1** | bottle | run | 0.19380157 | 0.35294117647058800 | 0.8425647690436400 | 0.78050363 |
| **1** | house | run | 0.49696347 | 0.8333333333333330 | 0.9034234796245880 | 0.87831223 |
| **2** | drum | health | -0.0077056717 | 0.3157894736842110 | 0.5563925773978070 | 0.9067832 |
| **2** | drum | milk | 0.08721569 | 0.7777777777777780 | 0.5975480176686130 | 0.9736985 |
| **2** | health | milk | 0.35118803 | 0.3157894736842110 | 0.6634753713833010 | 0.91598296 |
| **3** | doc | queen | 0.092088275 | 0.9166666666666670 | 0.562078996167126 | 0.9460243 |
| **3** | doc | friend | 0.28238988 | 0.8333333333333330 | 0.7200374126405000 | 0.6631671 |
| **3** | queen | friend | 0.48006862 | 0.8333333333333330 | 0.7307981474642620 | 0.7541387 |
| **4** | save | help | 0.7064365 | 0.9473684210526320 | 0.8547973240768060 | 0.8912734 |
| **4** | save | waste | 0.36981145 | 0.6666666666666670 | 0.7645042165335950 | 0.903338 |
| **4** | help | waste | 0.37867028 | 0.631578947368421 | 0.7202843082428950 | 0.6582974 |

**Commands:**

python3 p2\_ywang340.py input.csv output.csv score.csv

Table

Description automatically generated