SEMANTIC WORD COMPARATOR PROJECT

ALGORYTHMICS AND DATA MANAGEMENT

Liam Phan

Description of the program

sub-functions:

read_reference_text()
make_word_vector()
product()
similarity()

main function:

similar_word_computation()
main()

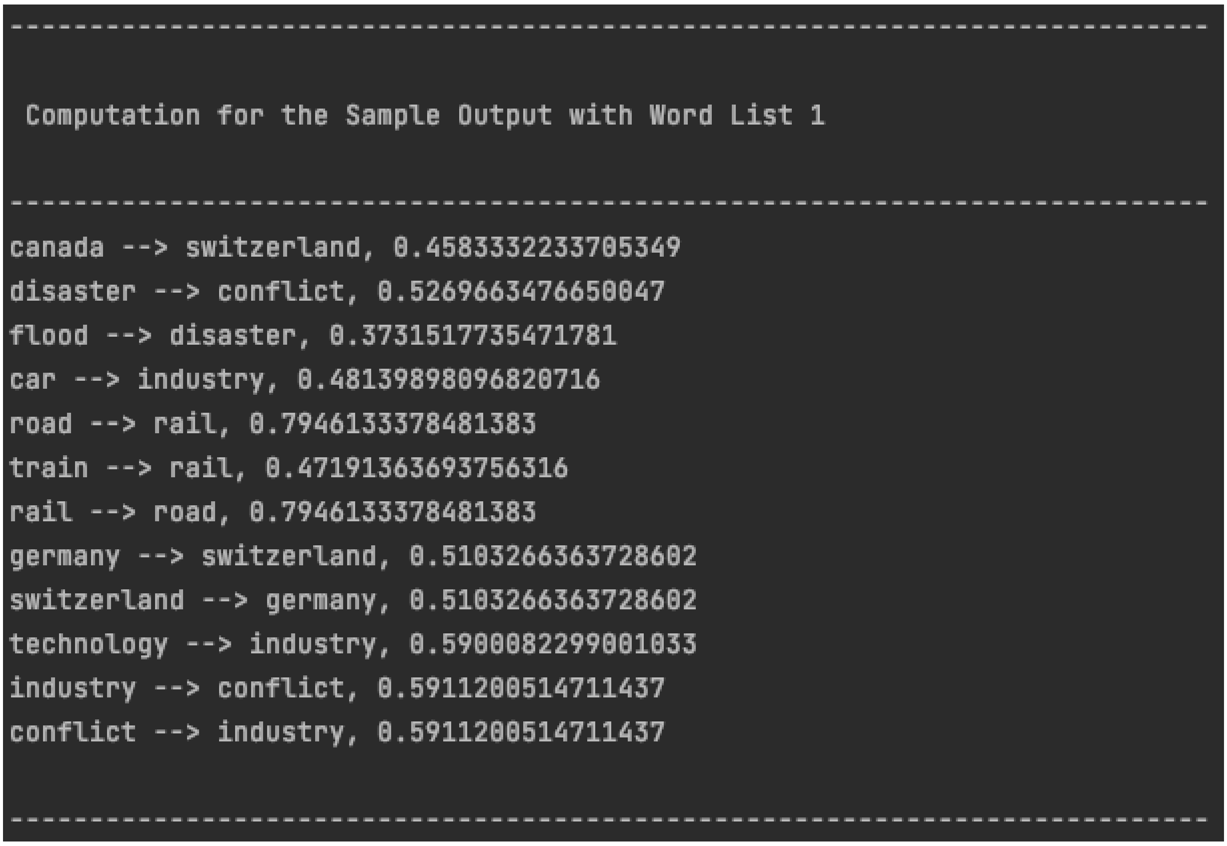
My program is broken down into 4 sub-functions and 1 main function grouping everything together. The output is generated with the **main()** to print the text and the requested values correctly.

Once a list of words is inserted, a reference text, a set of stopwords and an encoding type are filled in, the main function generates the cosine similarity of the words of the list thanks to the reference text, with a value oscillating between 0 and 1, 1 being the maximum of semantic similarity, often implying that the word is compared to itself, 0 the minimum implying that the word could not be compared with another word because of an impossible vector product between dictionaries as no contextual information exists.

My output is slightly different from the sample given (0.05 differences), notably with T**rain --> Road** which has become **Train --> Rail**. This may be due to the **read_reference_text()** function being slightly different in the way it separates lines and handles punctuation. But on the other hand it seems more logical, as **Train --> Rail** should be semantically closer than **Train --> Road**.

This work is mostly my own work and reflection, however I did get help with some of the concepts in the **make_word_vector function()** from **Michael Bigler**.

Screenshots of the Program Output

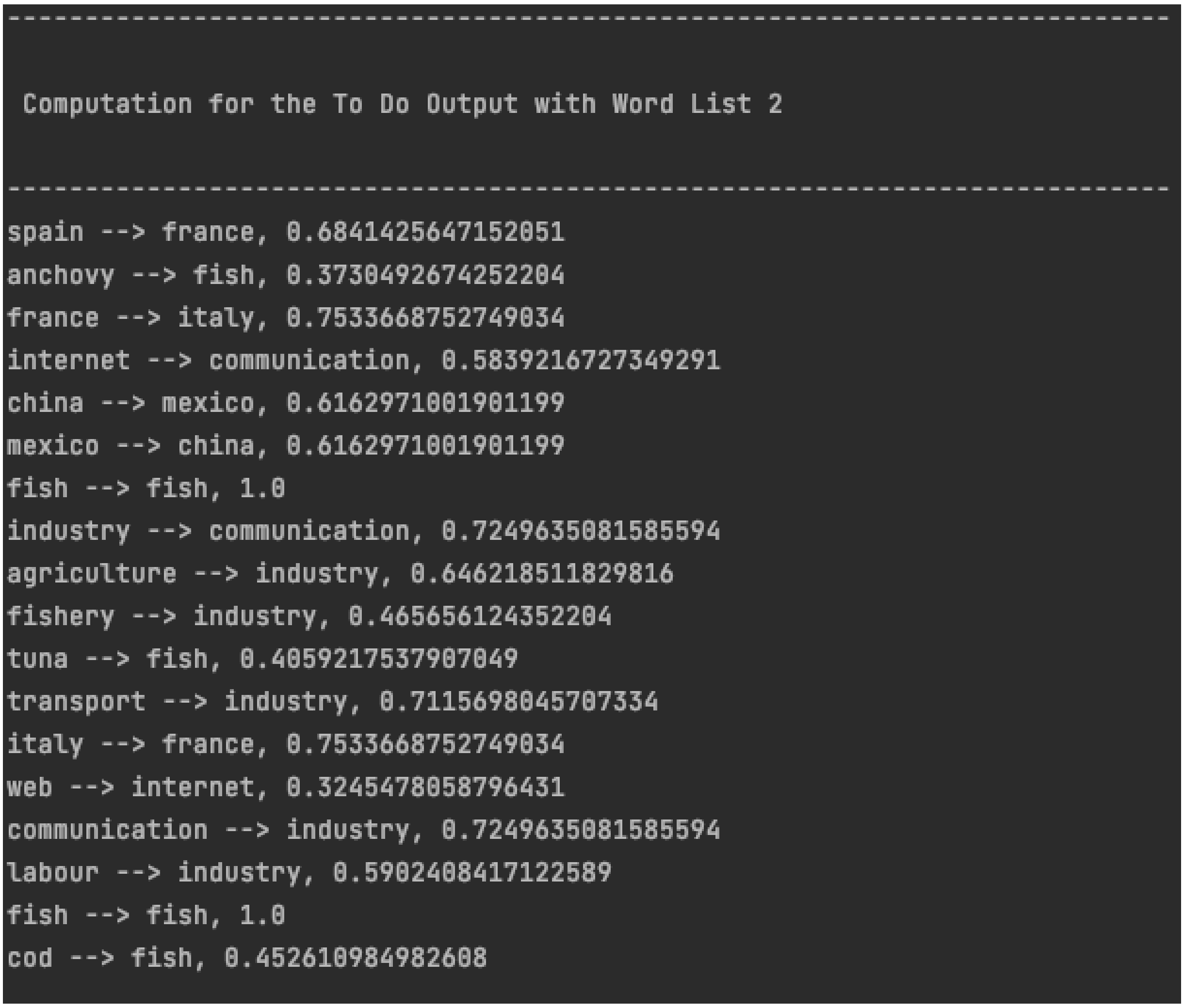


```
---------------------------------------------------------------------------


 Computation for the Sample Output with Word List 1


---------------------------------------------------------------------------
canada --> switzerland, 0.4583332233705349
disaster --> conflict, 0.5269663476650047
flood --> disaster, 0.3731517735471781
car --> industry, 0.48139898096820716
road --> rail, 0.7946133378481383
train --> rail, 0.47191363693756316
rail --> road, 0.7946133378481383
germany --> switzerland, 0.5103266363728602
switzerland --> germany, 0.5103266363728602
technology --> industry, 0.5900082299001033
industry --> conflict, 0.5911200514711437
conflict --> industry, 0.5911200514711437


---------------------------------------------------------------------------
```



```
---------------------------------------------------------------------------


 Computation for the To Do Output with Word List 2


---------------------------------------------------------------------------
spain --> france, 0.6841425647152051
anchovy --> fish, 0.3730492674252204
france --> italy, 0.7533668752749034
internet --> communication, 0.5839216727349291
china --> mexico, 0.6162971001901199
mexico --> china, 0.6162971001901199
fish --> fish, 1.0
industry --> communication, 0.7249635081585594
agriculture --> industry, 0.646218511829816
fishery --> industry, 0.465656124352204
tuna --> fish, 0.4059217537907049
transport --> industry, 0.7115698045707334
italy --> france, 0.7533668752749034
web --> internet, 0.3245478058796431
communication --> industry, 0.7249635081585594
labour --> industry, 0.5902408417122589
fish --> fish, 1.0
cod --> fish, 0.452610984982608
```