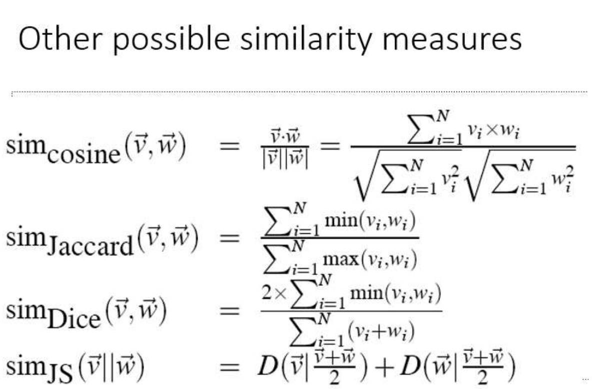
Extra credit – Tried to Improve Prediction Acuuracy

ATTEMPT -1 : Alternate similarity metrics

Implemented Jacard and Dice Similarity (Bad Accuracies)



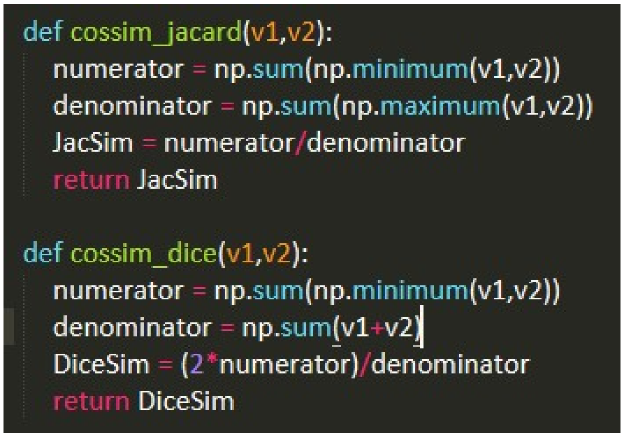
Other possible similarity measures

$$\mathrm{sim}_{\mathrm{cosine}}(\vec{v},\vec{w}) = \frac{\vec{v}\cdot\vec{w}}{|\vec{v}||\vec{w}|} = \frac{\sum_{i=1}^{N} v_i \times w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

$$\mathrm{sim}_{\mathrm{Jaccard}}(\vec{v},\vec{w}) = \frac{\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} \max(v_i,w_i)}$$

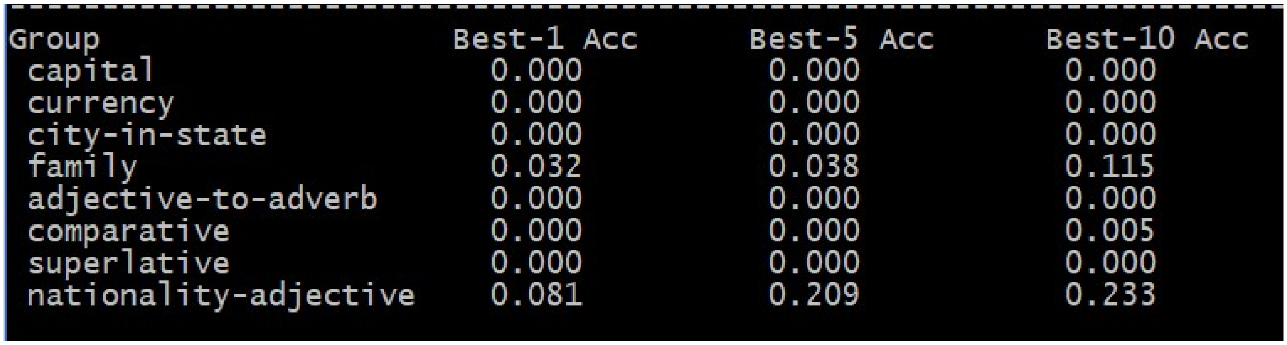
$$\mathrm{sim}_{\mathrm{Dice}}(\vec{v},\vec{w}) = \frac{2\times\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} (v_i+w_i)}$$

$$\mathrm{sim}_{\mathrm{JS}}(\vec{v}||\vec{w}) = D(\vec{v}|\tfrac{\vec{v}+\vec{w}}{2}) + D(\vec{w}|\tfrac{\vec{v}+\vec{w}}{2})$$



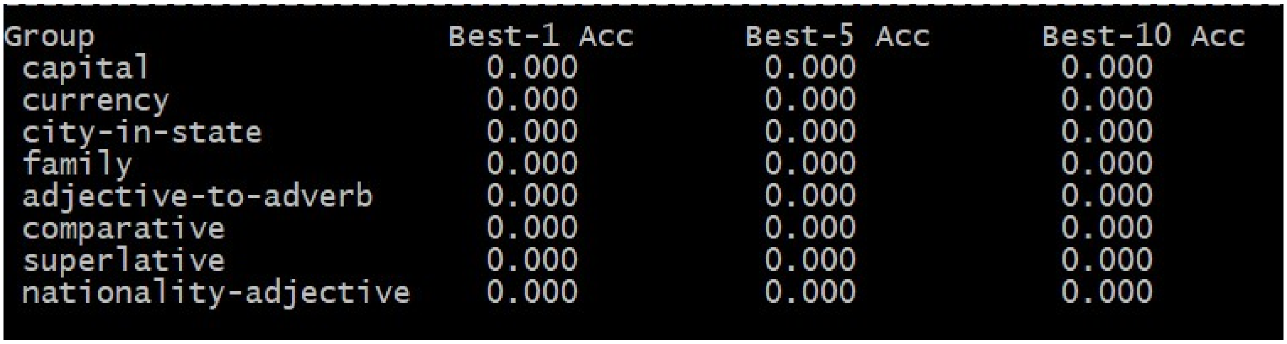
```
def cossim_jacard(v1,v2):
    numerator = np.sum(np.minimum(v1,v2))
    denominator = np.sum(np.maximum(v1,v2))
    JacSim = numerator/denominator
    return JacSim

def cossim_dice(v1,v2):
    numerator = np.sum(np.minimum(v1,v2))
    denominator = np.sum(v1+v2)
    DiceSim = (2*numerator)/denominator
    return DiceSim
```

Poor performance using Jacard similarity – not helpful



```
--------------------------------------------------------------------------------------------
Group                    Best-1 Acc      Best-5 Acc      Best-10 Acc
 capital                   0.000           0.000           0.000
 currency                  0.000           0.000           0.000
 city-in-state             0.000           0.000           0.000
 family                    0.032           0.038           0.115
 adjective-to-adverb       0.000           0.000           0.000
 comparative               0.000           0.000           0.005
 superlative               0.000           0.000           0.000
 nationality-adjective     0.081           0.209           0.233
```

Poor performance using Dice similarity – not helpful



```
--------------------------------------------------------------------------------------------
Group                    Best-1 Acc      Best-5 Acc      Best-10 Acc
 capital                   0.000           0.000           0.000
 currency                  0.000           0.000           0.000
 city-in-state             0.000           0.000           0.000
 family                    0.000           0.000           0.000
 adjective-to-adverb       0.000           0.000           0.000
 comparative               0.000           0.000           0.000
 superlative               0.000           0.000           0.000
 nationality-adjective     0.000           0.000           0.000
```

ATTEMPT - 2 : Alternate pre trained vectors

Used pre trained word vectors from https://nlp.stanford.edu/projects/glove/

Download the vectors for words from Wikipedia 2014 and Gigaword5(822 MB)

System lacked the processing power(process kept terminating abruptly)

ATTEMPT – 3 : Stemming words

Tried to add the context of word stem for word vectors. Logic: words having the same stem are more likely to have the same semantic meaning. However, this technique does not generate the words that can be derived or inflected from the stemmed forms. Hence did not improve prediction accuracy as expected.