For 2/16

Fix Tables

Part 4

Bag of words (find sorted order given endpoints: ignore ==)

Selected closest word that wasn’t the same word. For quartiles, selected also first word of different root.

**First Table (for each endpoint find closest word)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Word Pairs** |  |  |  |  |  |
| good:best | 'great': 0.711 | 'terrific: 0.693 | **'better': 0.645** | 'fantastic': 0.642 | 'excellent':0.638 |
| Good |  |  |  |  |  |
| best |  |  |  |  |  |
| bad:worst | 'terrible': 0.725 | 'horrible': 0.706 | 'dreadful': 0.651 | 'horrendous', 0.642 | 'lousy': 0.621 |
| slow:slowest | **'slower': 0.774** | 'slowed': 0.657 | 'fastest': 0.643 | 'painfully\_slow': 0.632 | 'slowing', 0.631 |
| fast:fastest | 'quickest': 0.703 | 'slowest': 0.674 | **'faster':0.646** | 'slower’: 0.559 | 'speediest': 0.542 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Word Pairs** |  |  |  |  |  |
| Happy:happiest | **'happier', 0.694** | 'luckiest', 0.61 | ‘overjoyed’, 0.577 | unhappiest, 0.568 | proudest, 0.567 |
| Sad:saddest | 'saddening', 0.667 | 'heartbreaking', 0.663 | **'sadder', 0.629** | 'tragic', 0.627 | Meny\_Friedman', 0.619 |
| Angry:angriest | 'irate', 0.671 | 'enraged', 0.631 | **'angrier', 0.613** | 'furious', 0.587 | 'indignant', 0.585 |

Sample code (Not sure whether to add a negative):

Good = vecs.get\_vector(“good”)

Best = vects.get\_vector(“best”)

goodbest = best + ((good-best)/2)

middlegoodbest = vecs.most\_similar(positive=[goodbest], topn=10);

**Second Table (List 10 nearest neighbors for each endpoint, quarter**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **First Word** | **1st Quarter** | **Half** | **2nd Quarter** | **Second Word** |
| **Furious 0.999** | 'livid', 0.734 | unhappy, 0.701 | 'glad', 0.715 | **Happy 0.999** |
| **Furious 0.999** | 'angry', 0.686  'livid', 0.660 | 'angry', 0.593  'frantic', 0.577 | 'calmed', 0.662  'quiet', 0.627 | **Calm 0.999** |
| **Terrible 1** | 'horrible', 0.894  'horrendous', 0.810 | 'horrible', 0.792  'great', 0.778 | 'fantastic', 0.826 | **Terrific 1** |
| **Cold 1** | 'chilly', 0.693  'frigid', 0.673 | 'chilly', 0.645  'frigid', 0.616  'warm', 0.601 | 'cold', 0.667  'CHEFS\_Chefs', 0.586  'hotter', 0.578 | **Hot 1** |
| **Ugly** | 'uglier', 0.651  'hideous', 0.649 | 'beautiful', 0.729  'lovely', 0.673 | 'beautiful', 0.828  'lovely', 0.782 | **Gorgeous 1** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **First Word** | **1st Quarter** | **Half** | **2nd Quarter** | **Second Word** |
| **Black** | 'Responded\_Letterman\_How', 0.629 | 'brown', 0.648  'blue', 0.6346  'Responded\_Letterman\_How', 0.625 | 'brown', 0.662  'blue', 0.656  'wrote\_Newitz', 0.622 | **White** |
| **Dark** | 'darkened', 0.61623  'gray', 0.589 | 'ghostly\_glow', 0.596 | 'yellowish\_glow', 0.590 | **Light** |
| **Sad** | 'saddening', 0.682 | 'glad', 0.6997 | 'glad', 0.7486 | **Happy** |

a = vecs.get\_vector(**"furious"**)  
b = vecs.get\_vector(**"calm"**)  
c = b/2 + a/2

c1 = 3b/4 + a/4

c3 = b/4 + (3\*a/4)  
m = vecs.most\_similar(positive=[c], topn=3)  
print(m)