# Word Similarity

Mohammed Muqeeth[$EE16B026$] and Vaibhav Nayel[$EE15B157$]

IIT Madras

## 1 Knowledge Based Methods

### 1.1 Explicit semantic analysis

Wikipedia is the largest knowledge source known. The content in Wikipedia covers many fields like history, logic,politics, culture and arts etc .Therefore we used Explicit semantic technique with the knowledge of Wikipedia. The dump of articles in English Wikipedia till 2018 is of 15 GB which on extraction into XML results in 70 GB. In ESA, each word is represented as a vector(u) in space spanned by articles of Wikipedia. We compute relatedness between words(vectors u and v) using cosine similarity.

**Implementation** Considering the computational resources , We took only 4 GB of Wikipedia. Using online resource we made raw text in json format with articles having minimum of 100 words and incoming and outgoing links greater than or equal to 5. This data of size 1.8 GB has about 0.2 million articles with about 0.6 million words. The stop words in text are removed. The text is then tokenized and lemmatizer from wordnet is used to convert each token into its root form for latest english wikipedia and simple wikipedia.The simple wikipedia is then stemmed using porter stemmer. We created indexing for each article and each token. We constructed a semantic interpreter which is term-document matrix. The general tf-idf scheme is used in construction of term-document matrix.

Each row in term-document matrix corresponds to a word represented in articles space.The cosine formula is used for computation of similarity between two words which are vectors in articles space.

$$sim_{cosine}(w1, w2) = \frac{\sum_i w1_i w2_i}{\sqrt{\sum_i w1_i^2}\sqrt{\sum_i w1_i^2}}$$

**Analysis:** We analysed how our ESA model works with different relations like antonyms,synonyms between word pairs.The following tables shows these results with ESA model built using simple-wiki articles and latest english articles.

| | Word 1 | Word 2 | Latest article wikipedia simple englsih wiki | |
|---|---|---|---|---|
| **0** | privacy | covert | 0.0289379 | 0 |
| **1** | walker | runner | 0.010654 | 0.00567317 |
| **2** | dream | horror | 0.0325022 | 0.0120172 |
| **3** | speech | discourse | 0.0775112 | 0.0264023 |
| **4** | temper | mood | 0.0364456 | 0.00334731 |
| **5** | scholastic | academic | 0.072163 | 0.0121005 |
| **6** | refer | cite | 0.0961133 | 0.154913 |

| | Word 1 | Word 2 | Latest articles wikipedia simple english wiki | |
|---|---|---|---|---|
| **0** | pain | joy | 0.0178071 | 0.0106496 |
| **1** | culprit | victim | 0.0249695 | 0.0218297 |
| **2** | witch | natural | 0.0163153 | 0.0155007 |
| **3** | artificial | actual | 0.0719689 | 0.0447051 |
| **4** | allude | refer | 0.0394987 | 0.0101064 |
| **5** | enmity | amity | 0.00985834 | 0 |
| **6** | atheist | theist | 0.0855875 | 0.592268 |
| **7** | impeccable | fallible | 0 | 0 |

| | Word 1 | Word 2 | Latest english wikipedia | simple english wiki |
|---|---|---|---|---|
| **0** | car | wheel | 0.269206 | 0.0979711 |
| **1** | airplane | wheel | 0.0289009 | 0.00515624 |
| **2** | airplane | fuel | 0.0784663 | 0.0600542 |
| **3** | tree | bark | 0.162982 | 0.290848 |
| **4** | human | hand | 0.131934 | 0.0856357 |
| **5** | hand | finger | 0.194095 | 0.252639 |
| **6** | human | finger | 0.0441385 | 0.0251186 |
| **7** | chair | leg | 0.0529404 | 0.173073 |
| **8** | human | leg | 0.0647619 | 0.0641657 |

| | Word 1 | Word 2 | Latest english wikipedia | simple english wiki |
|---|---|---|---|---|
| **0** | animal | dog | 0.154651 | 0.149163 |
| **1** | dog | labrador | 0.124256 | 0.0547719 |
| **2** | animal | labrador | 0.0148247 | 0.00644028 |
| **3** | red | crimson | 0.0416337 | 0.0531848 |
| **4** | angle | acute | 0.0228004 | 0.0256484 |
| **5** | material | metal | 0.166705 | 0.106856 |
| **6** | material | steel | 0.156238 | 0.119867 |

In all tables ESA based on latest english articles gives more relatedness between word pairs compared to ESA model on simple wiki because of difficult word pairs we have chosen.This also tells us that if we have large wikipedia dump similarity between difficult words can be estimated. Some of the word pairs in above tables have zero similarity which implies they do not occur together in a document. ESA works good with synonyms,meronyms and hypernyms because such word pairs occur together in a given article. Antonyms generally do not occur together in the same article therefore similarity values are less compared to other relations.

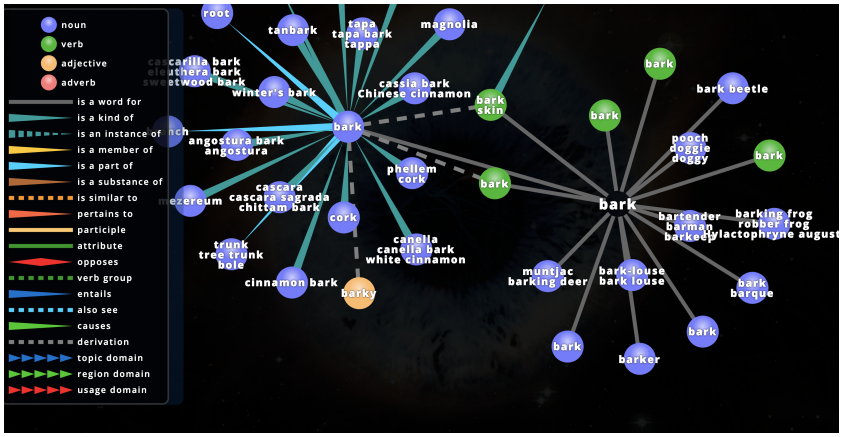
The tf-idf scheme used in our model is vulnerable to keyword spammers(ie a word appears repeatedly in an article).We can come up with new schemes such as normalized frequency for calculating weights in term-document matrix.If stemming is done for latest english articles also, performance would have been better.

**Evaluation:** We computed word similarity between word pairs in wordsim-353 corpus. Spearman's rank correlation is used as a measure for evaluation.We

obtained correlation of 0.455 and 0.555 for ESA based on english wikipedia and simple-wiki respectively.

### 1.2 WordNet

WordNet is a lexical network whose basic unit is the synset. A synset is a set of synonyms or words representing the same concept. The synsets are linked to each other through various lexical relations. Wordnet consists of 117,000 synsets which fall into 4 large groups: nouns, verbs, adjectives and pronouns. There are 10 noun relations, 6 verb relations, 5 adjective relations, and 2 adverb relations. Relations are exemplified by antonyms, meronyms, hypernyms, etc. These relations were coded by hand. The following graphic shows how synsets may be linked.




In order to measure similarity between two words, we must examine the ways in which they are connected through this lexical graph. While similarity measures have been proposed for wordnet, we chose to implement 4 of them in this report.

**Leacock-Chodorow:** This is an edge-counting measure defined as follows:

$$sim_{LC} = -\log(\frac{l(c_1, c_2)}{2L})$$

Where L is the depth of the taxonomy, which is 30 in our case and it only works with nouns.

**Wu-Palmer:** This is also an edge counting technique. It is different from LC because is takes into account the depth of the words being consider, or more precisely, their most specific common ancestor.

$$sim_{WP} = \frac{2N_3}{2N_3 + N_1 + N_2}$$

where $N_1$ and $N_2$ are the distances of the 2 words to their lowest ancestor and $N_3$ is the depth of the lowest common ancestor.

**Liu:** The Liu measure is very similar to WP in that it takes into account depth of the lowest superclass , however it allows us to trade-off how much importance is given to the depth and the distance of the 2 words from each other using parameters $\alpha$ and *beta*.

$$sim_{Liu} = e^{-\alpha L} \frac{e^{\beta H} - e^{-\beta H}}{e^{\beta H} + e^{-\beta H}}$$

where $L = N1 + N2$ and $H = N_3$ used in the WP measure.

**Adapted Lesk:** The Lesk measure is a corpus based approach to the similarity problem. It uses a corpus to create word vectors of the words appearing in the glosses of the 2 words being considered. These vectors are added to make a document vector. The correlation between these 2 document vectors gives the similarity between words.

**Analysis:** We created 4 sets of word pairs with 10 rows each in order to observe what kind of patterns each of our models learns. The 4 sets are meronyms, hypernyms, synonyms and antonyms.

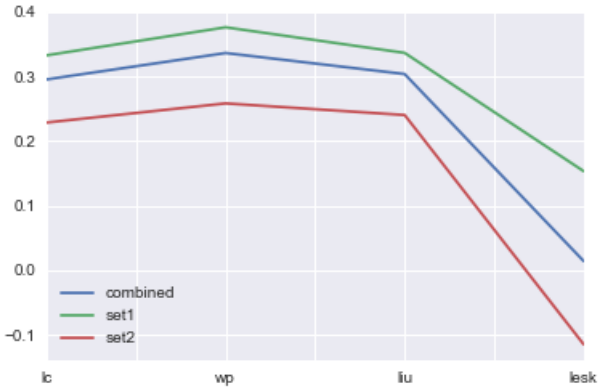
| | Word 1 | Word 2 | lc | wp | liu | lesk |
|---|---|---|---|---|---|---|
| 0 | privacy | covert | 1.23969 | 0.285714 | 4.53999e-05 | 0.897911 |
| 1 | walker | runner | 2.53897 | 0.869565 | 0.135335 | 0.881748 |
| 2 | dream | horror | 1.84583 | 0.666667 | 0.00673795 | 0.486147 |
| 3 | speech | discourse | 2.94444 | 0.933333 | 0.367879 | 0.66602 |
| 4 | temper | mood | 3.63759 | 1 | 1 | 0.858284 |
| 5 | hazy | obscure | 0 | 0 | 0 | nan |
| 6 | scholastic | academic | 1.55814 | 0.571429 | 0.000911882 | 0.740682 |
| 7 | misfortune | mischance | 2.94444 | 0.933333 | 0.367879 | 0.407306 |
| 8 | energy | briskness | 2.53897 | 0.875 | 0.135335 | 0.335581 |
| 9 | refer | cite | 0 | 1 | 1 | 0.89688 |

| | Word 1 | Word 2 | lc | wp | liu | lesk |
|---|---|---|---|---|---|---|
| 0 | pain | joy | 2.25129 | 0.769231 | 0.0497871 | 0.892946 |
| 1 | culprit | victim | 2.02815 | 0.666667 | 0.0183156 | 0.511294 |
| 2 | witch | natural | 2.02815 | 0.666667 | 0.0183156 | 0.888125 |
| 3 | artificial | actual | 0 | 0 | 0 | 0.337714 |
| 4 | latent | explicit | 0 | 0 | 0 | nan |
| 5 | allude | refer | 0 | 0.166667 | 0 | 0.865998 |
| 6 | enmity | amity | 2.02815 | 0.666667 | 0.0183156 | 0.851712 |
| 7 | zeal | lethargy | 0 | 0 | 0 | nan |
| 8 | atheist | theist | 1.69168 | 0.6 | 0.00247875 | 0.872025 |
| 9 | impeccable | fallible | 0 | 0 | 0 | 0.741374 |

| | Word 1 | Word 2 | lc | wp | liu | lesk |
|---|---|---|---|---|---|---|
| **0** | car | wheel | 2.53897 | 0.8 | 0.135335 | 0.903413 |
| **1** | airplane | wheel | 1.69168 | 0.727273 | 0.00247875 | 0.746614 |
| **2** | car | fuel | 0 | 0 | 0 | nan |
| **3** | airplane | fuel | 0.998529 | 0.235294 | 2.26033e-06 | 0.553672 |
| **4** | tree | bark | 1.335 | 0.470588 | 0.00012341 | 0.584668 |
| **5** | human | hand | 0.998529 | 0.48 | 2.26033e-06 | 0.901597 |
| **6** | hand | finger | 2.53897 | 0.875 | 0.135335 | 0.968871 |
| **7** | human | finger | 0.998529 | 0.380952 | 2.26033e-06 | 0.878288 |
| **8** | chair | leg | 2.25129 | 0.842105 | 0.0497871 | 0.931881 |
| **9** | human | leg | 0.929536 | 0.363636 | 8.31529e-07 | 0.928568 |

| | Word 1 | Word 2 | lc | wp | liu | lesk |
|---|---|---|---|---|---|---|
| **0** | animal | dog | 2.53897 | 0.875 | 0.135335 | 0.878455 |
| **1** | dog | labrador | 1.23969 | 0.375 | 4.53999e-05 | 0.898952 |
| **2** | animal | labrador | 1.44036 | 0.428571 | 0.000335463 | 0.912987 |
| **3** | colour | red | 0 | 0 | 0 | nan |
| **4** | red | crimson | 2.94444 | 1 | 0.367879 | 0.978327 |
| **5** | colour | crimson | 0 | 0 | 0 | nan |
| **6** | angle | acute | 0.998529 | 0.4 | 2.26033e-06 | 0.902659 |
| **7** | material | metal | 2.25129 | 0.615385 | 0.0497871 | 0.915328 |
| **8** | metal | steel | 0 | 0 | 0 | nan |
| **9** | material | steel | 2.02815 | 0.705882 | 0.0183156 | 0.818852 |

LC appears to understand synonym and antonym relations quite well but it loses out meronyms and hypernyms as the degrees of separation increase between them (for example animal and labrador).
WP seems to correlate highly with LC and suffers from the same problems.
In our experiments, Liu seemed to be very sensitive to the parameters alpha and beta and hence we were not able to achieve good results with it.
Lesk is by far the worst of all the similarity measures. Its output has roughly zero correlation indicating that it is mostly random. This is likely because glosses don't give enough information about each word form.
Finally, the following correlation coefficients were obtained on wordsim353 for each of the methods.

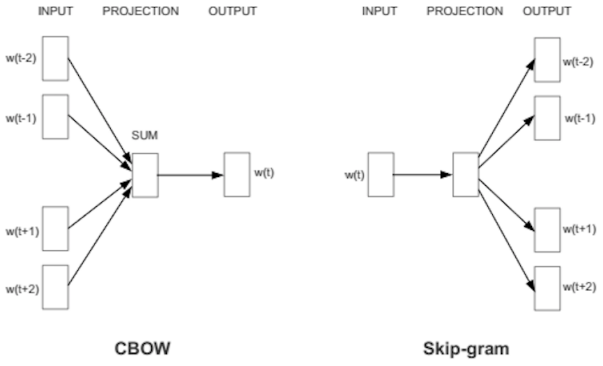



Wu Palmer is the best similarity we found on wordnet.

## 2 Corpus based methods

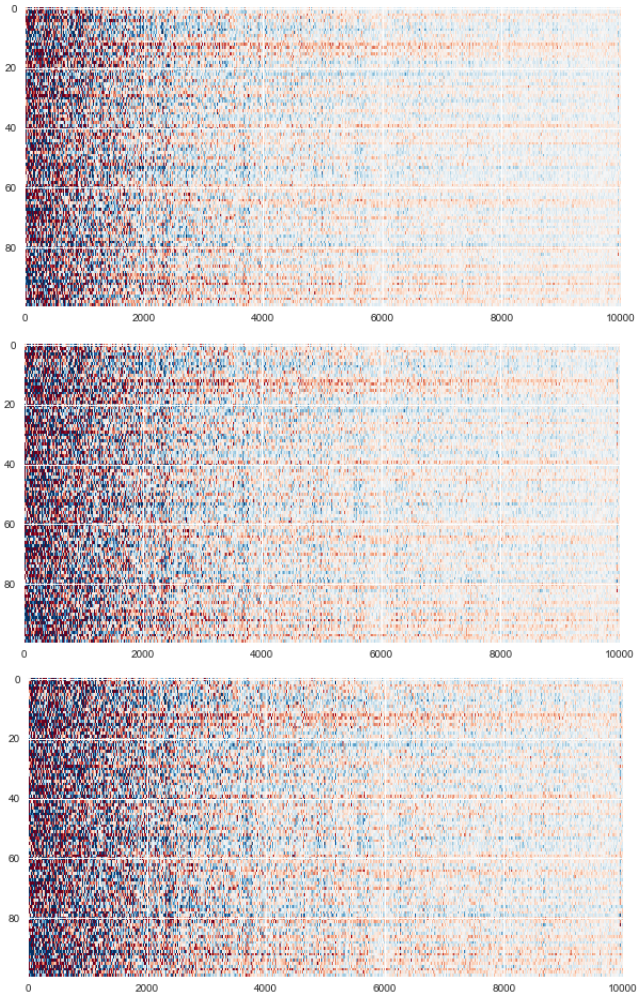
### 2.1 Word2Vec

Word2vec is a neural network used to create word embeddings using large corpora. The network consists of 2 layers (hidden and output) and is trained in one of two ways: CBOW and skip-gram. skip-gram attempts to predict the context words of the given word and CBOW (continuous bag of words) does the reverse task of predicting the missing word given its context words in a window around it. In both cases, the word embedding is taken to be the hidden layer representation.




There are several parameters which can be tuned in this model such as the number of hidden units, number of epochs trained and the training method(CBOW or skip-gram).

**Analysis:** To see the effect of training for different epochs, we use the reuters dataset which has a vocabulary of 14k words and plot the most frequent word vectors through the epochs at intervals of 2. In this example, the model has 100 hidden units and is trained with CBOW.




We clearly see the word vectors becoming more dense as the training progresses. The more infrequent words to the right become more dense as the network recognizes them better over time.
Now let us see the effect of epochs on the word similarities predicted by the model.

| | Word 1 | Word 2 | w2v_12 | w2v_2 | w2v_4 | w2v_6 |
|---|---|---|---|---|---|---|
| 0 | privacy | covert | nan | 0.616944 | 0.47791 | 0.427325 |
| 1 | walker | runner | nan | 0.396043 | 0.288376 | 0.304937 |
| 2 | dream | horror | nan | 0.608583 | 0.560614 | 0.522876 |
| 3 | speech | discourse | nan | 0.449588 | 0.360185 | 0.373166 |
| 4 | temper | mood | nan | 0.703566 | 0.619496 | 0.603228 |
| 5 | hazy | obscure | nan | nan | nan | nan |
| 6 | scholastic | academic | nan | 0.542605 | 0.592898 | 0.580085 |
| 7 | misfortune | mischance | nan | nan | nan | nan |
| 8 | energy | briskness | nan | nan | nan | nan |
| 9 | refer | cite | nan | 0.390228 | 0.357132 | 0.337591 |

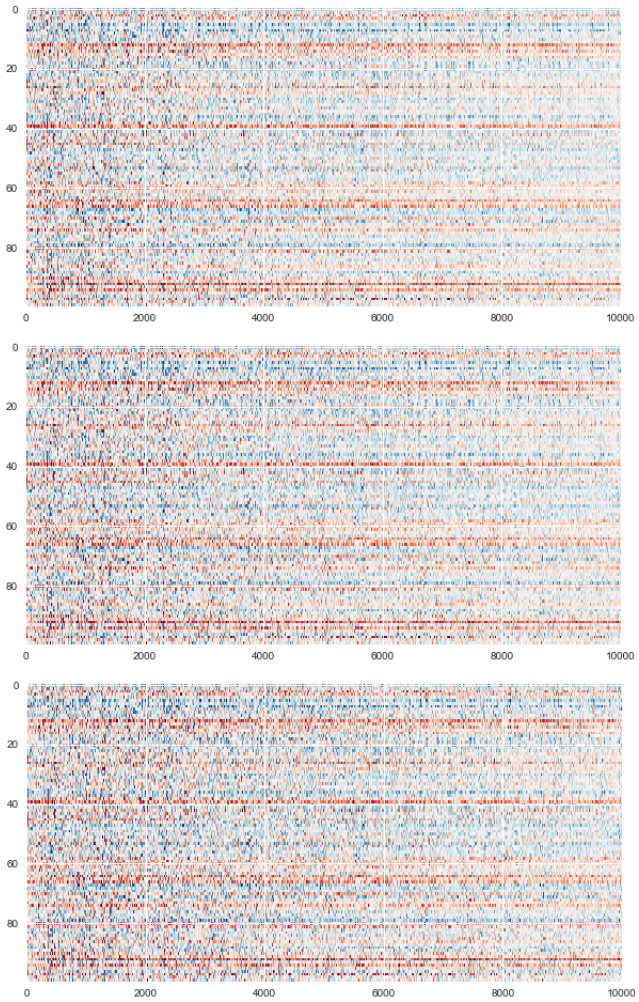
| | Word 1 | Word 2 | w2v_12 | w2v_2 | w2v_4 | w2v_6 |
|---|---|---|---|---|---|---|
| 0 | pain | joy | nan | 0.239638 | 0.21343 | 0.219117 |
| 1 | culprit | victim | nan | 0.468539 | 0.348728 | 0.341095 |
| 2 | witch | natural | nan | -0.114055 | -0.0946639 | -0.0850246 |
| 3 | artificial | actual | -0.00175807 | 0.466903 | 0.36338 | 0.33777 |
| 4 | latent | explicit | nan | nan | nan | nan |
| 5 | allude | refer | nan | 0.568952 | 0.519245 | 0.515644 |
| 6 | enmity | amity | nan | 0.118575 | 0.236527 | 0.201953 |
| 7 | zeal | lethargy | nan | nan | nan | nan |
| 8 | atheist | theist | nan | 0.438047 | 0.297768 | 0.381405 |
| 9 | impeccable | fallible | nan | nan | nan | nan |

| | Word 1 | Word 2 | w2v_12 | w2v_2 | w2v_4 | w2v_6 |
|---|---|---|---|---|---|---|
| 0 | car | wheel | nan | 0.636923 | 0.618935 | 0.563279 |
| 1 | airplane | wheel | nan | 0.546315 | 0.460962 | 0.397731 |
| 2 | car | fuel | nan | nan | nan | nan |
| 3 | airplane | fuel | nan | 0.553936 | 0.469178 | 0.446122 |
| 4 | tree | bark | nan | 0.799534 | 0.751277 | 0.723264 |
| 5 | human | hand | 0.431095 | 0.29502 | 0.239355 | 0.241854 |
| 6 | hand | finger | nan | 0.721371 | 0.666501 | 0.651473 |
| 7 | human | finger | nan | 0.210537 | 0.144393 | 0.129094 |
| 8 | chair | leg | nan | 0.4328 | 0.315338 | 0.315035 |
| 9 | human | leg | nan | 0.199005 | 0.130427 | 0.115251 |

| | Word 1 | Word 2 | w2v_12 | w2v_2 | w2v_4 | w2v_6 |
|---|---|---|---|---|---|---|
| 0 | animal | dog | nan | 0.591821 | 0.546343 | 0.521899 |
| 1 | dog | labrador | nan | 0.160705 | 0.255402 | 0.26299 |
| 2 | animal | labrador | nan | 0.122358 | 0.133631 | 0.125088 |
| 3 | colour | red | nan | nan | nan | nan |
| 4 | red | crimson | nan | 0.453536 | 0.355974 | 0.301178 |
| 5 | colour | crimson | nan | nan | nan | nan |
| 6 | angle | acute | nan | 0.480949 | 0.406145 | 0.349064 |
| 7 | material | metal | 0.518483 | 0.552042 | 0.475649 | 0.503527 |
| 8 | metal | steel | nan | nan | nan | nan |
| 9 | material | steel | 0.332896 | 0.553363 | 0.457839 | 0.416313 |

Here we can see that the similarity values are changing. where the network had previously over or underestimated the similarity, it corrected itself. However the more common words don't seem to change in similarity, probably because the network has already learned a good representation for them and it is not likely to change. This model tends not to understand meronym and hypernym pairs

since it is trained to recognize word neighbourhoods. Since the words 'human' and 'finger' are unlikely to occur surrounded by the same words, the model does not learn is-a and part-of relations. However, it is able to relate synonyms and antonyms relatively well.
Next we would like to see the difference between cbow and skipgram in terms of similarity evolution as well as vector density.




In general, the vector density is more spread out across the words unlike the cbow model where the density was more concentrated to the left of the graph i.e. towards more frequent words. The reason for this might be that cbow concentrates on predicting the missing word and concentrates more on frequent words since they are of course more likely to occur. Whereas skipgram tries to predict contexts given a single word and so does not face the problem of having to predict more frequent words. This is why the embedded vector appears more uniformly distributed over all the words.

| | Word 1 | Word 2 | w2v_sg_2 | w2v_sg_4 | w2v_sg_6 |
|---|---|---|---|---|---|
| 0 | privacy | covert | 0.834554 | 0.740452 | 0.643109 |
| 1 | walker | runner | 0.567336 | 0.405597 | 0.342377 |
| 2 | dream | horror | 0.691582 | 0.593572 | 0.548403 |
| 3 | speech | discourse | 0.628156 | 0.557806 | 0.485458 |
| 4 | temper | mood | 0.81538 | 0.734573 | 0.683435 |
| 5 | hazy | obscure | nan | nan | nan |
| 6 | scholastic | academic | 0.759049 | 0.708621 | 0.673116 |
| 7 | misfortune | mischance | nan | nan | nan |
| 8 | energy | briskness | nan | nan | nan |
| 9 | refer | cite | 0.396351 | 0.303183 | 0.26126 |

| | Word 1 | Word 2 | w2v_sg_2 | w2v_sg_4 | w2v_sg_6 |
|---|---|---|---|---|---|
| 0 | pain | joy | 0.477707 | 0.412462 | 0.409336 |
| 1 | culprit | victim | 0.770242 | 0.767406 | 0.750826 |
| 2 | witch | natural | 0.311758 | 0.216603 | 0.16597 |
| 3 | artificial | actual | 0.552141 | 0.448791 | 0.421533 |
| 4 | latent | explicit | nan | nan | nan |
| 5 | allude | refer | 0.488194 | 0.468305 | 0.444456 |
| 6 | enmity | amity | 0.911286 | 0.77434 | 0.659474 |
| 7 | zeal | lethargy | nan | nan | nan |
| 8 | atheist | theist | 0.725659 | 0.731214 | 0.717373 |
| 9 | impeccable | fallible | nan | nan | nan |

| | Word 1 | Word 2 | w2v_sg_2 | w2v_sg_4 | w2v_sg_6 |
|---|---|---|---|---|---|
| 0 | car | wheel | 0.667745 | 0.5921 | 0.568594 |
| 1 | airplane | wheel | 0.623102 | 0.489544 | 0.425633 |
| 2 | car | fuel | nan | nan | nan |
| 3 | airplane | fuel | 0.641028 | 0.549409 | 0.490304 |
| 4 | tree | bark | 0.800605 | 0.741982 | 0.727701 |
| 5 | human | hand | 0.531308 | 0.444431 | 0.412597 |
| 6 | hand | finger | 0.782687 | 0.746943 | 0.729596 |
| 7 | human | finger | 0.463499 | 0.401857 | 0.385234 |
| 8 | chair | leg | 0.524529 | 0.457607 | 0.470107 |
| 9 | human | leg | 0.445752 | 0.36874 | 0.340602 |

| | Word 1 | Word 2 | w2v_sg_2 | w2v_sg_4 | w2v_sg_6 |
|---|---|---|---|---|---|
| 0 | animal | dog | 0.657237 | 0.610686 | 0.594507 |
| 1 | dog | labrador | 0.549793 | 0.48627 | 0.428862 |
| 2 | animal | labrador | 0.532154 | 0.40869 | 0.359508 |
| 3 | colour | red | nan | nan | nan |
| 4 | red | crimson | 0.674049 | 0.577077 | 0.542693 |
| 5 | colour | crimson | nan | nan | nan |
| 6 | angle | acute | 0.498491 | 0.424309 | 0.417764 |
| 7 | material | metal | 0.5477 | 0.54086 | 0.523114 |
| 8 | metal | steel | nan | nan | nan |
| 9 | material | steel | 0.546371 | 0.446562 | 0.45897 |

Judging by the performance over our toy examples, it seems the skipgram has learned better representations since it assigns higher similarities than cbow model in nearly all cases. The only drawback here is that skipgram takes much

longer to train than cbow
Lastly we will look at the effect of changing the hidden layer size from 50 to 100. This is our final evaluation of word2vec so we will compare 4 models. The above 2 mentioned hidden layer sizes trained using CBOW and skip-gram.

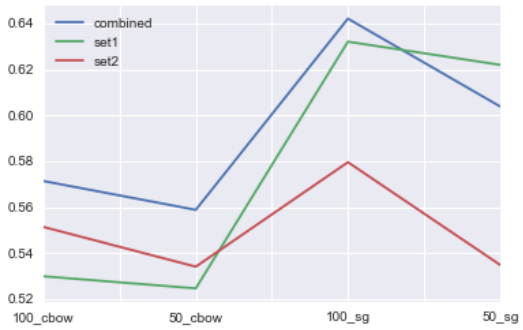
| | Word 1 | Word 2 | 100_cbow | 50_cbow | 100_sg | 50_sg |
|---|---|---|---|---|---|---|
| **0** | privacy | covert | 0.487483 | 0.480566 | 0.69187 | 0.75752 |
| **1** | walker | runner | 0.265486 | 0.32324 | 0.406392 | 0.437744 |
| **2** | dream | horror | 0.523943 | 0.576231 | 0.558763 | 0.651485 |
| **3** | speech | discourse | 0.352387 | 0.388286 | 0.538418 | 0.678977 |
| **4** | temper | mood | 0.661266 | 0.707268 | 0.705136 | 0.756911 |
| **5** | hazy | obscure | nan | nan | nan | nan |
| **6** | scholastic | academic | 0.54093 | 0.593281 | 0.668657 | 0.719076 |
| **7** | misfortune | mischance | nan | nan | nan | nan |
| **8** | energy | briskness | nan | nan | nan | nan |
| **9** | refer | cite | 0.326349 | 0.35493 | 0.298702 | 0.274368 |

| | Word 1 | Word 2 | 100_cbow | 50_cbow | 100_sg | 50_sg |
|---|---|---|---|---|---|---|
| **0** | pain | joy | 0.276028 | 0.189412 | 0.405234 | 0.398236 |
| **1** | culprit | victim | 0.378067 | 0.447848 | 0.779419 | 0.819203 |
| **2** | witch | natural | -0.0779653 | -0.00501256 | 0.169098 | 0.17768 |
| **3** | artificial | actual | 0.347763 | 0.394102 | 0.458419 | 0.529137 |
| **4** | latent | explicit | nan | nan | nan | nan |
| **5** | allude | refer | 0.332819 | 0.465128 | 0.406295 | 0.518882 |
| **6** | enmity | amity | 0.0795582 | 0.114754 | 0.725283 | 0.747767 |
| **7** | zeal | lethargy | nan | nan | nan | nan |
| **8** | atheist | theist | 0.346114 | 0.433668 | 0.730697 | 0.747249 |
| **9** | impeccable | fallible | nan | nan | nan | nan |

| | Word 1 | Word 2 | 100_cbow | 50_cbow | 100_sg | 50_sg |
|---|---|---|---|---|---|---|
| 0 | car | wheel | 0.604337 | 0.66257 | 0.557991 | 0.709613 |
| 1 | airplane | wheel | 0.379394 | 0.498853 | 0.39974 | 0.588637 |
| 2 | car | fuel | nan | nan | nan | nan |
| 3 | airplane | fuel | 0.457997 | 0.544024 | 0.550287 | 0.676379 |
| 4 | tree | bark | 0.745399 | 0.799819 | 0.72184 | 0.861211 |
| 5 | human | hand | 0.220952 | 0.243264 | 0.457597 | 0.535577 |
| 6 | hand | finger | 0.66659 | 0.73932 | 0.760553 | 0.846753 |
| 7 | human | finger | 0.1474 | 0.188167 | 0.405335 | 0.474427 |
| 8 | chair | leg | 0.319145 | 0.337387 | 0.485185 | 0.535953 |
| 9 | human | leg | 0.119841 | 0.132243 | 0.348132 | 0.460524 |

| | Word 1 | Word 2 | 100_cbow | 50_cbow | 100_sg | 50_sg |
|---|---|---|---|---|---|---|
| 0 | animal | dog | 0.489096 | 0.584391 | 0.605041 | 0.634975 |
| 1 | dog | labrador | 0.248182 | 0.224767 | 0.48599 | 0.607376 |
| 2 | animal | labrador | 0.140887 | 0.136185 | 0.378135 | 0.478108 |
| 3 | colour | red | nan | nan | nan | nan |
| 4 | red | crimson | 0.280901 | 0.351446 | 0.592856 | 0.685424 |
| 5 | colour | crimson | nan | nan | nan | nan |
| 6 | angle | acute | 0.34748 | 0.401761 | 0.43541 | 0.470449 |
| 7 | material | metal | 0.494233 | 0.494633 | 0.55039 | 0.662795 |
| 8 | metal | steel | nan | nan | nan | nan |
| 9 | material | steel | 0.470903 | 0.494757 | 0.483888 | 0.590696 |

The correlations given by these models are plotted in the graph below:



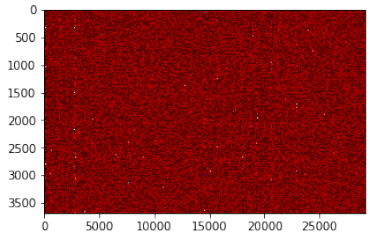

More hidden units seem to perform better than fewer. This is likely due to more degrees of freedom available to the model in order to represent a larger variety of words.

## 2.2 Latent semantic analysis

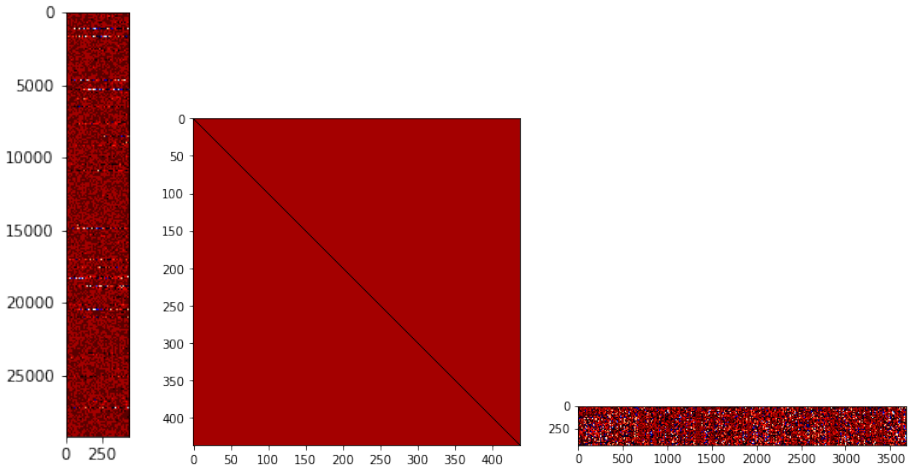
Explicit semantic analysis assumes every article as independent of other articles.We resolve this by extraction of information in term-document matrix to concept representation. We use singular value decomposition to achieve this.After

svd we can represent each word and article in concept space.The corpus used here is simple wiki english articles.

svd decomposes term-dcoument matrix A into three matrices.Here we used term-document matrix for 1000 articles. Taking first k singular values gives us representation shown below.




(a) term-document matrix after svd



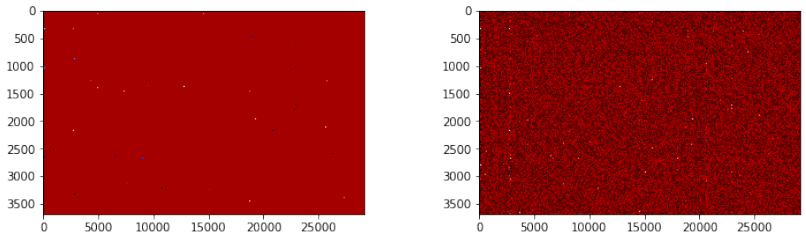

(b) column orthogonal matrix$U$

(c) diagonal Matrix$\Sigma$

(d) row orthogonal matrix$V$

**Implementation** Matrix computed in ESA model is used for svd decomposition.After choosing first k singular values.The low rank approximation of term-document matrix is made by multiplying the three matrices.The similarity between word pair is calculated in the same way as in ESA.

**Analysis:** The following table shows LSA implementation with two different ranks k=500 , k=1000 and k = 100




(a) term-document matrix before svd

(b) term-document matrix after svd

Fig. 1: Figure showing how svd makes term-document matrix denser

| | Word 1 | Word 2 | lsa rank = 500 | lsa rank = 1000 | lsa rank = 100 |
|---|---|---|---|---|---|
| **0** | privacy | covert | 0.14108 | 0.03884 | 0.21811 |
| **1** | walker | runner | 0.05271 | 0.01109 | 0.22234 |
| **2** | dream | horror | 0.06532 | 0.09053 | 0.24125 |
| **3** | speech | discourse | 0.13711 | 0.09296 | 0.43297 |
| **4** | temper | mood | 0.06824 | 0.03485 | 0.13622 |
| **5** | scholastic | academic | 0.21963 | 0.15987 | 0.37664 |
| **6** | refer | cite | 0.19002 | 0.01621 | 0.19709 |

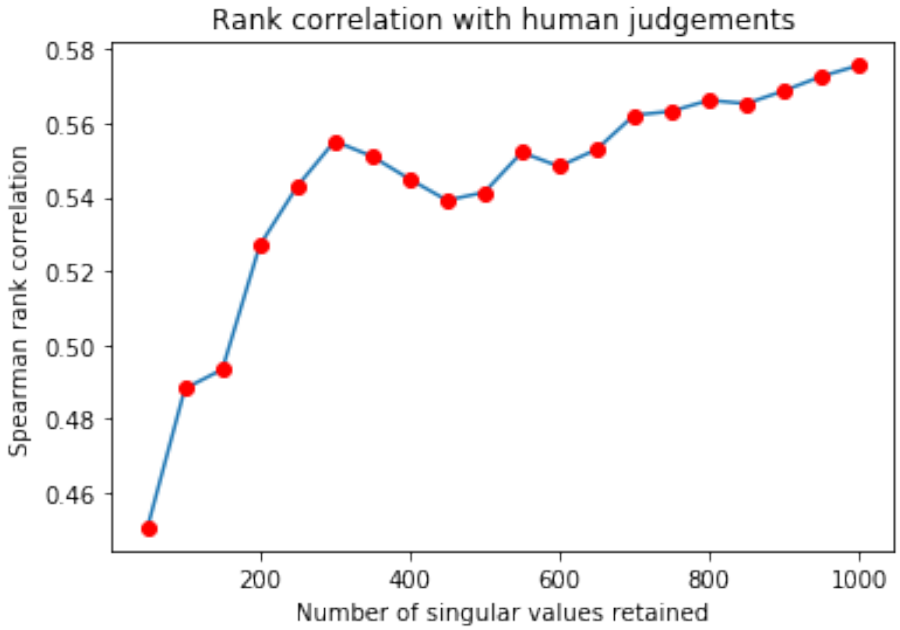
| | Word 1 | Word 2 | lsa rank = 500 | Lsa rank = 1000 | lsa rank = 100 |
|---|---|---|---|---|---|
| **0** | pain | joy | 0.07064 | 0.04091 | 0.13542 |
| **1** | culprit | victim | 0.12019 | 0.11767 | 0.16345 |
| **2** | witch | natural | 0.04354 | -0.00129 | 0.18629 |
| **3** | artificial | actual | 0.22209 | 0.10029 | 0.49012 |
| **4** | allude | refer | 0.03751 | 0.04409 | 0.07584 |
| **5** | enmity | amity | 0.04094 | 0.02352 | 0.03595 |
| **6** | atheist | theist | 0.92493 | 0.9477 | 0.8467 |

| | Word 1 | Word 2 | lsa rank =500 | lsa rank = 1000 | lsa rank = 100 |
|---|---|---|---|---|---|
| **0** | car | wheel | 0.14558 | 0.10367 | 0.70914 |
| **1** | airplane | wheel | 0.0693 | 0.01842 | 0.28046 |
| **2** | airplane | fuel | 0.19697 | 0.13178 | 0.55888 |
| **3** | tree | bark | 0.80023 | 0.77368 | 0.93759 |
| **4** | human | hand | 0.21403 | 0.12899 | 0.40619 |
| **5** | hand | finger | 0.48247 | 0.4266 | 0.50596 |
| **6** | human | finger | 0.07309 | 0.0565 | 0.25053 |
| **7** | chair | leg | 0.53495 | 0.43902 | 0.77228 |
| **8** | human | leg | 0.09112 | 0.08832 | 0.20003 |

| | Word 1 | Word 2 | lsa rank = 500 | lsa rank = 1000 | lsa rank = 100 |
|---|---|---|---|---|---|
| **0** | animal | dog | 0.23274 | 0.17516 | 0.81813 |
| **1** | dog | labrador | 0.6431 | 0.45588 | 0.2695 |
| **2** | animal | labrador | 0.10217 | 0.0454 | 0.33606 |
| **3** | red | crimson | 0.15506 | 0.08376 | 0.31171 |
| **4** | angle | acute | 0.01328 | 0.0225 | 0.05717 |
| **5** | material | metal | 0.29616 | 0.1288 | 0.65306 |
| **6** | material | steel | 0.39769 | 0.22477 | 0.46518 |

Some of word pairs like (dog,labrador),(chair,leg),(tree,bark) have high similarity compared to that of esa approach because they fall into same concepts. LSA understands synonyms,hypernyms better because such word pairs fall under the same concepts. For some antonyms pair such as witch,natural lsa gives negative similarity with value close to zero.Lsa is not giving better performance on antonyms because these pairs do not fall in same concepts. For ranks as low as 100 the similarities between synonyms,hypernyms are high since concepts are very low.Even for low ranks antonyms seem to perform worse. meronyms on low rank are performing better when compared to that of high ranks.

**Evaluation** The decomposition is done for different values of k ranging from 50 to 1000 in steps of 50.The plot of spearman rank correlation with rank k is shown in Fig. 2.2).

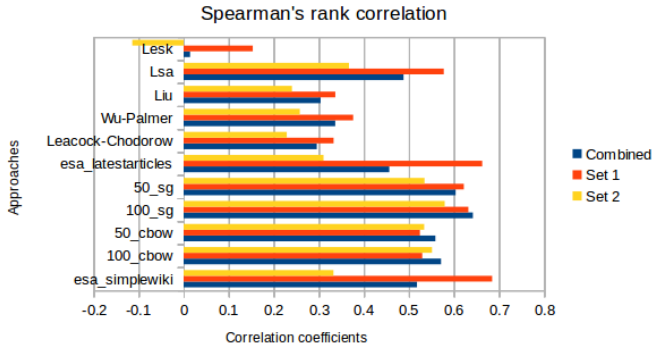



## 3 Conclusion

Below we tabulate out qualitative analysis of the techniques applied. We tried to assess whether each model had learned the relationships contained in each of the 4 sets of words we chose.

| | w2v | lc | wp | liu | lesk | lsa | esa |
|---|---|---|---|---|---|---|---|
| syn | y | y | y | n | n | y | y |
| ant | y | y | y | n | y | n | n |
| mer | y | n | y | n | n | n | y |
| hyp | y | y | n | n | n | y | y |

Below is a quantitative comparison of the our models on 3 partitions of the wordsim 353 dataset. Word2vec performs best on the combined dataset.


Spearman's rank correlation
Lesk
Lsa
Liu
Wu-Palmer
Leacock-Chodorow
esa_latestarticles
50_sg
100_sg
50_cbow
100_cbow
esa_simplewiki
Approaches
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Correlation coefficients
Combined
Set 1
Set 2