1 Gold Standard

One large issue that we had was choosing an appropriate gold standard for cluster evaluation. From our evaluation SimLex-999 is the dataset having the largest and most exact task of estimating semantic similarity between words and avoiding relatedness. Table 1 shows the difference between SimLex-999 and WordSim-353.

2 Other Word Embeddings

For GloVe we used pretrained 200 dimensional vector embeddings\(^1\) trained using Wikipedia 2014 + Gigaword 5 (6B tokens). Eigenwords were trained on English Gigaword with no lowercasing or cleaning. Finally, we used 50 dimensional vector representations from Huang et al. (2012), which used the April 2010 snapshot of the Wikipedia corpus (Lin, 1998; Shaoul, 2010), with a total of about 2 million articles and 990 million tokens.

In table 2 we show a qualitative comparison between multiple word embedding. We show that many word embeddings contain antonyms, and also that thesauri include rare words and rare senses. It should be noted that signed clustering can easily be applied to word sense aware embeddings and thesauri.

3 Further Cluster Evaluation

Next we evaluated our clusters using an external gold standard. Cluster purity and entropy (Zhao and Karypis, 2001) is defined as,

\[
Purity = \sum_{r=1}^{k} \frac{1}{n} \max_{i} (n_i^r) \\
Entropy = \sum_{r=1}^{k} \frac{n_r}{n} \left( -\frac{1}{\log q} \sum_{i=1}^{q} \frac{n_i^r}{n_r} \log \frac{n_i^r}{n_r} \right)
\]

where \(q\) is the number of classes, \(k\) the number of clusters, \(n_r\) the size of cluster \(r\), and \(n_i^r\) number of data points in class \(i\) clustered in cluster \(r\). The purity and entropy measures improve (increased purity, decreased entropy) monotonically with the number of clusters.

The number of disconnected components (NDC) in the cluster where we only use synonym edges.

\[
NDC = \sum_{r=1}^{k} \sum_{i=1}^{C} (n_i^r)
\]

4 Hyperparameter Optimization

We show the optimization results using 10-fold cross validation on our 5108 word dataset. The optimal hyperparameters are chosen by minimizing error, as seen in table 3.

Table 3 shows out of sample results from the grid search of hyperparameter optimization. Here we show that Eigenword + MSW outperforms Eigenword + Roget, which is in contrast with the other word embeddings where the combination with Roget performs better. Another interesting result from the hyperparameter optimization is that Word2Vec with Roget has two very different optima.

5 Explanded Results

When compared with the MS Word thesaurus, Word2Vec, Eigenword, GloCon, and GloVe word
Table 1: Comparison between SimLex-999 and WordSim-353. This is from http://www.cl.cam.ac.uk/~fh295/simlex.html.

<table>
<thead>
<tr>
<th>Pair</th>
<th>SimLex-999 rating</th>
<th>WordSim-353 rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>coast - shore</td>
<td>9.00</td>
<td>9.10</td>
</tr>
<tr>
<td>clothes - closet</td>
<td>1.96</td>
<td>8.00</td>
</tr>
</tbody>
</table>

Table 2: Qualitative comparison of clusters.

Table 3: Clustering evaluation after parameter optimization minimizing error using grid search.

Table 4: Sentiment analysis accuracy for binary predictions of signed clustering algorithm (SC) versus other models.

Embeddings had a total of 286, 235, 235, 220 negative edges, respectively. The results are similar with the other thesauri.

If we examined the number of disconnected components within the different word clusters, we observed that when K-means were used, the number of disconnected components were statistically significant from random labelling. This suggests that the word embeddings capture synonym relationships. By optimizing the hyperparameters we found roughly a 10 percent decrease in disconnected components using normalized cuts. When we added the signed antonym relationships using our signed clustering algorithm, on average we found a 39 percent decrease over the K-means clusters.


References


Dekang Lin. 1998. Automatic retrieval and clustering