Semantic Word Clusters Using Signed Spectral Clustering

João Sedoc, Jean Gallier, Lyle Ungar
Computer & Information Science
University of Pennsylvania
joao, jean, ungar@cis.upenn.edu

Dean Foster
Amazon LLC
dean@foster.net

Abstract

Vector space representations of words capture many aspects of word similarity, but such methods tend to produce vector spaces in which antonyms (as well as synonyms) are close to each other. For spectral clustering using such word embeddings, words are points in a vector space where synonyms are linked with positive weights, while antonyms are linked with negative weights. We present a new signed spectral normalized graph cut algorithm, *signed clustering*, that overlays existing thesauri upon distributionally derived vector representations of words, so that antonym relationships between word pairs are represented by negative weights. Our signed clustering algorithm produces clusters of words that simultaneously capture distributional and synonym relations. By using randomized spectral decomposition (Halko et al., 2011) and sparse matrices, our method is both fast and scalable. We validate our clusters using datasets containing human judgments of word pair similarities and show the benefit of using our word clusters for sentiment prediction.

1 Introduction

In distributional vector representations, opposite relations are not fully captured. Take, for example, words such as "great" and "awful" that can appear with similar frequency in the same sentence structure: "John had a great meeting" and "John had an awful day." Word embeddings, which are successful in a wide array of NLP tasks (Turney et al., 2010; Dhillon et al., 2015), fail to capture this antonymy because they follow the *distributional hypothesis* that similar words are used in similar contexts (Harris, 1954), thus assigning small cosine or euclidean distances between the vector representations of "great" and "awful".

While vector space models (Turney et al., 2010) such as word2vec (Mikolov et al., 2013), Global vectors (GloVe) (Pennington et al., 2014), or Eigenwords (Dhillon et al., 2015) capture relatedness, they do not adequately encode synonymy and semantic similarity (Mohammad et al., 2013; Scheible et al., 2013). Our goal is to create clusters of synonyms or semantically equivalent words and linguistically motivated unified constructs. Signed graphs, which are graphs with negative edge weights, were first introduced by Cartwright and Harary (1956). However, signed graph clustering for multiclass normalized cuts (K-clusters) has been largely unexplored until recently. We present a novel theory and method that extends multiclass normalized cuts (K-cluster) of Yu and Shi (2003) to signed graphs (Gallier, 2016)[1] and the work of Kunegis et al. (2010) to K-clustering. This extension allows the incorporation of knowledge base information, positive and negatively weighted links (see figure 2.1). Negative edges serve as repellent or opposite relationships between nodes.

Our signed spectral normalized graph cut algorithm (henceforth, signed clustering) builds negative edge relations into graph embeddings using similarity structure in vector spaces. It takes as input an initial set of vectors and edge relations, and hence is easy to combine with any word embedding method. This paper formally improves on the discrete optimization problem of Yu and Shi (2003).

Signed clustering gives better clusters than spectral clustering (Shi and Malik, 2000) of word embeddings, and it has better coverage and is more robust than thesaurus look-up. This is because the-

[1] Gallier (2016) is a full theoretical exposition of our methods with proofs on arXiv.

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pages 939–949
Vancouver, Canada, July 30 - August 4, 2017. ©2017 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17-1087

sauri erroneously give equal weight to rare senses of a word – for example, “rich” as a rarely used synonym of “absurd”. Also, the overlap between thesauri is small, due to their manual creation. Lin (1998) found 17.8397% overlap between synonym sets from Roget’s Thesaurus and WordNet 1.5. We find similarly small overlap between all three thesauri tested.

We evaluate our clusters using SimLex-999 (Hill et al., 2014) and SimVerb-3500 (Gerz et al., 2016) as a ground truth for our cluster evaluation. Finally, we test our method on the sentiment analysis task. Overall, signed spectral clustering can augment methods using signed information and has broad application for many fields.

Our main contributions are: the novel extension of signed clustering to the multiclass (K-cluster), and the application of this method to create semantic word clusters that are agnostic to vector space representations and thesauri.

1.1 Related Work

Semantic word cluster and distributional thesauri have been well studied in the NLP literature (Lin, 1998; Curran, 2004). Recently there has been a line of research on incorporating synonyms and antonyms into word embeddings. Our approach is very much in the line of Vlachos et al. (2009). However, they explicitly made verb clusters using Dirichlet Process Mixture Models and must-link / cannot-link clustering. Furthermore, they note that cannot-link clustering does not improve performance whereas our signed clustering antonyms are key.

Most recent models either attempt to make richer contexts, in order to find semantic similarity, or overlay thesaurus information in a supervised or semi-supervised manner. One line of active research is post processing the word vector embedding by transforming the space using a single or multi-relational objective (Yih et al., 2012; Tang et al., 2014; Chang et al., 2013; Tang et al., 2014; Zhang et al., 2014; Faruqui et al., 2015; Mrkšić et al., 2016).

Alternatively, there are methods to modify the objective function for generating the word embeddings (Ono et al., 2015; Pham et al., 2015; Schwartz et al., 2015).

Our approach differs from the aforementioned methods in that we created word clusters using the antonym relationships as negative links. Unlike the previous approaches using semi-supervised methods, we incorporated the thesauri as a knowledge base. Similar to word vector retrofitting and counter-fitting methods described in Faruqui et al. (2015) and Mrkšić et al. (2016), our signed clustering method uses existing vector representations to create word clusters.

To our knowledge, this work is the first theoretical foundation of multiclass signed normalized cuts.[2] Zass and Shashua (2005) solved multiclass cluster from another approach, by relaxing the orthogonality assumption and focusing instead on the non-negativity constraint. This led to a doubly stochastic optimization problem. Negative edges are handled by a constrained hyperparameter. Hou (2005) used positive degrees of nodes in the degree matrix of a signed graph with weights (-1, 0, 1), which was advanced by Kolluri et al. (2004) and Kunegis et al. (2010) using absolute values of weights in the degree matrix. Interestingly, Chiang et al. (2014) presented a theoretical foundation for edge sign prediction and a recursive clustering approach. Mercado et al. (2016) found that using the geometric mean of the graph Laplacian improves performance.

Wang et al. (2016) used semi-supervised polarity induction (Rao and Ravichandran, 2009) to create clusters of words with similar valence and arousal. Must-link and cannot-link soft spectral clustering (Rangapuram and Hein, 2012) share similarities with our method, particularly in the limit where there are no must-link edges present. Both must-link and cannot-link clustering as well as polarity induction differ in optimization method. Our method is significantly faster due to the use of randomized SVD (Halko et al., 2011) and can thus be applied to large scale NLP problems.

We developed a novel theory and algorithm that extends the clustering of Shi and Malik (2000) and Yu and Shi (2003) to the multiclass signed graph case.

2 Signed Graph Cluster Estimation

2.1 Signed Normalized Cut

Weighted graphs for which the weight matrix is a symmetric matrix in which negative and positive entries are allowed are called *signed graphs*.

[2] The full exposition by Gallier (2016) is available on arXiv.

Such graphs (with weights $(-1, 0, +1)$) were introduced as early as 1953 by (Harary, 1953), to model social relations involving disliking, indifference, and liking. The problem of clustering the nodes of a signed graph arises naturally as a generalization of the clustering problem for weighted graphs. Figure 1 shows a signed graph of word similarities with a thesaurus overlay. Gallier (2016) extends normalized cuts to signed graphs in order to incorporate antonym information into word clusters.

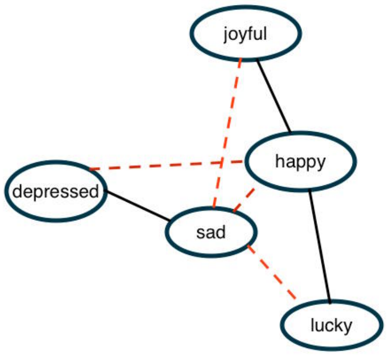

Figure 1: Signed graph of words using a distance metric from the word embedding. The red dashed edges represent the antonym relation while solid edges represent synonymy relations.

Definition 2.1. A *weighted graph* is a pair $G = (V, W)$, where $V = \{v_1, \ldots, v_m\}$ is a set of *nodes* or *vertices*, and W is a symmetric matrix called the *weight matrix*, such that $w_{i\,j} \geq 0$ for all $i, j \in \{1, \ldots, m\}$, and $w_{i\,i} = 0$ for $i = 1, \ldots, m$. We say that a set $\{v_i, v_j\}$ is an edge iff $w_{i\,j} > 0$. The corresponding (undirected) graph (V, E) with $E = \{\{v_i, v_j\} \mid w_{i\,j} > 0\}$, is called the *underlying graph* of G.

Given a signed graph $G = (V, W)$ (where W is a symmetric matrix with zero diagonal entries), the *underlying graph* of G is the graph with node set V and set of (undirected) edges $E = \{\{v_i, v_j\} \mid w_{ij} \neq 0\}$.

If (V, W) is a signed graph, where W is an $m \times m$ symmetric matrix with zero diagonal entries and with the other entries $w_{ij} \in \mathbb{R}$ arbitrary, for any node $v_i \in V$, the *signed degree* of v_i is defined as

$$\overline{d}_i = \overline{d}(v_i) = \sum_{j=1}^{m} |w_{ij}|,$$

and the *signed degree matrix* $\overline{D}$ as

$$\overline{D} = \operatorname{diag}(\overline{d}(v_1), \ldots, \overline{d}(v_m)).$$

For any subset A of the set of nodes V, let

$$\operatorname{vol}(A) = \sum_{v_i \in A} \overline{d}_i = \sum_{v_i \in A} \sum_{j=1}^{m} |w_{ij}|.$$

For any two subsets A and B of V and A^C which is the complement of A, define $\operatorname{links}^{+}(A, B)$, $\operatorname{links}^{-}(A, B)$, and $\operatorname{cut}(A, A^C)$ by

$$\operatorname{links}^{+}(A, B) = \sum_{\substack{v_i \in A, v_j \in B \\ w_{ij} > 0}} w_{ij}$$

$$\operatorname{links}^{-}(A, B) = \sum_{\substack{v_i \in A, v_j \in B \\ w_{ij} < 0}} -w_{ij}$$

$$\operatorname{cut}(A, A^C) = \sum_{\substack{v_i \in A, v_j \in A^C \\ w_{ij} \neq 0}} |w_{ij}|.$$

Then, the *signed Laplacian* $\overline{L}$ is defined by

$$\overline{L} = \overline{D} - W,$$

and its normalized version $\overline{L}_{\text{sym}}$ by

$$\overline{L}_{\text{sym}} = \overline{D}^{-1/2}\, \overline{L}\, \overline{D}^{-1/2} = I - \overline{D}^{-1/2} W \overline{D}^{-1/2}.$$

Kunegis et al. (2010) showed that $\overline{L}$ is positive semidefinite. For a graph without isolated vertices, we have $\overline{d}(v_i) > 0$ for $i = 1, \ldots, m$, so $\overline{D}^{-1/2}$ is well defined.

Given a partition of V into K clusters $(A_1, \ldots, A_K)$, if we represent the jth block of this partition by a vector X^j such that

$$X_i^j = \begin{cases} a_j & \text{if } v_i \in A_j \\ 0 & \text{if } v_i \notin A_j, \end{cases}$$

for some $a_j \neq 0$. For illustration, suppose $m = 5$ and $A_1 = \{v_1, v_3\}$ then $(X^1)^\top = [a_1, 0, a_1, 0, 0]$.

Definition 2.2. The *signed normalized cut* $\operatorname{sNcut}(A_1, \ldots, A_K)$ of the partition $(A_1, ..., A_K)$ is defined as

$$\operatorname{sNcut}(A_1, \ldots, A_K) = \sum_{j=1}^{K} \frac{\operatorname{cut}(A_j, A_j^C) + 2\operatorname{links}^{-}(A_j, A_j)}{\operatorname{vol}(A_j)}.$$

It should be noted that this formulation differs significantly from Kunegis et al. (2010) and even more so from must-link / cannot-link clustering.

Observe that minimizing $\mathrm{sNcut}(A_1, \ldots, A_K)$ minimizes the number of positive and negative edges between clusters and also the number of negative edges within clusters. Removing the term $\mathrm{links}^-(A_j, A_j)$ reduces sNcut to normalized cuts.

A linear algebraic formulation is

$$\mathrm{sNcut}(A_1, \ldots, A_K) = \sum_{j=1}^{K} \frac{(X^j)^\top \overline{L} X^j}{(X^j)^\top \overline{D} X^j}.$$

where X is the $N \times K$ matrix whose jth column is X^j.

2.2 Optimization Problem

We now formulate K-way clustering of a graph using normalized cuts.

If we let

$$\mathcal{X} = \Big\{ [X^1 \; \ldots \; X^K] \mid X^j = a_j(x_1^j, \ldots, x_N^j), \\ x_i^j \in \{1, 0\}, a_j \in \mathbb{R}, \; X^j \neq 0 \Big\}$$

our solution set is

$$\mathcal{K} = \Big\{ X \in \mathcal{X} \mid (X^i)^\top \overline{D} X^j = 0, \\ 1 \leq i, j \leq K, \quad i \neq j \Big\}.$$

The resulting optimization problem is

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{j=1}^{K} \frac{(X^j)^\top \overline{L} X^j}{(X^j)^\top \overline{D} X^j} \\ \text{subject to} & (X^i)^\top \overline{D} X^j = 0, \\ & 1 \leq i, j \leq K, \; i \neq j, \quad X \in \mathcal{X}. \end{array}$$

The problem can be reformulated to an equivalent optimization problem:

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}(X^\top \overline{L} X) \\ \text{subject to} & X^\top \overline{D} X = I, \quad X \in \mathcal{X}. \end{array}$$

We then form a relaxation of the above problem, dropping the condition that $X \in \mathcal{X}$, giving

Relaxed Problem

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}(Y^\top \overline{D}^{-1/2} \overline{L}\,\overline{D}^{-1/2} Y) \\ \text{subject to} & Y^\top Y = I. \end{array}$$

The minimum of the relaxed problem is achieved by the K unit eigenvectors associated with the smallest eigenvalues of L_{sym}.

2.3 Finding an Approximate Discrete Solution

Given a solution Z of the relaxed problem, we look for pairs (X, Q) with $X \in \mathcal{X}$ and where Q is a $K \times K$ matrix with nonzero and pairwise orthogonal columns, with $\|X\|_F = \|Z\|_F$, that minimize

$$\varphi(X, Q) = \|X - ZQ\|_F \, .$$

Here, $\|A\|_F$ is the Frobenius norm of A.

This nonlinear optimization problem involves two unknown matrices X and Q. To solve the relaxed problem, we proceed by alternating between minimizing $\varphi(X, Q) = \|X - ZQ\|_F$ with respect to X holding Q fixed (step 5 in algorithm 1), and minimizing $\varphi(X, Q)$ with respect to Q holding X fixed (steps 6 and 7 in algorithm 1).

This second stage in which X is held fixed has been studied, but it is still a hard problem for which no closed-form solution is known. Hence we divide the problem into steps 6 and 7 for which the solution is known. Since Q is of the form $Q = R\Lambda$ where $R \in \mathbf{O}(K)$ and Λ is a diagonal invertible matrix, we minimize $\|X - ZR\Lambda\|_F$. The matrix $R\Lambda$ is not a minimizer of $\|X - ZR\Lambda\|_F$ in general, but it is an improvement on R alone, and both stages can be solved quite easily. In step 6 the problem reduces to minimizing $-2\mathrm{tr}(Q^\top Z^\top X)$; that is, maximizing $\mathrm{tr}(Q^\top Z^\top X)$.

Algorithm 1 Signed Clustering

```
1: Input: $W$ the weight matrix (without isolated nodes), K the number of clusters, and termination threshold $\epsilon$.
2: Using the $\overline{D}$ the degree matrix, and the signed Laplacian $\overline{L}$, compute $\overline{L}_{sym}$ the signed normalized Laplacian.
3: Initialize $\Lambda = I$, $X = \overline{D}^{-\frac{1}{2}} U$ where $U$ is the matrix of the eigenvectors corresponding to the K smallest eigenvalues of $\overline{L}_{sym}$.[3]
4: while $\|X - ZR\Lambda\|_F > \epsilon$ do
5:     Minimize $\|X - ZR\Lambda\|_F$ with respect to $X$ holding $Q$ fixed.
6:     Fix $X$, $Z$, and $\Lambda$, find $R \in \mathbf{O}(K)$ that minimizes $\|X - ZR\Lambda\|_F$.
7:     Fix $X$, $Z$, and $R$, find a diagonal invertible matrix $\Lambda$ that minimizes $\|X - ZR\Lambda\|_F$.
8: end while
9: Find the discrete solution $X^*$ by choosing the largest entry $x_{ij}$ on row $i$ set $x_{ij} = 1$ and all other $x_{ij} = 0$ for row $i$.
10: Output: $X^*$.
```

Steps 3 through 10 may be replaced by standard K-means clustering. It should also be noted that by removing the solution requirement that $X^j \neq 0$, the algorithm can find $k \leq K$ clusters.

3 Similarity Calculation

The main input to the spectral signed clustering algorithm is the similarity matrix W, which overlays both the distributional properties and thesaurus information. Following Belkin and Niyogi (2003), we chose the heat kernel based on the Euclidean distance between word vector representations as our similarity metric, such that

$$W_{ij} = \begin{cases} 0 & \text{if } e^{-\frac{\|w_i - w_j\|^2}{\sigma}} < \epsilon \\ e^{-\frac{\|w_i - w_j\|^2}{\sigma}} & \text{otherwise} \end{cases}.$$

where σ and ϵ are hyperparameters found using grid search (see Supplemental material for more detail).

We represented the thesaurus as two matrices where

$$T^{syn}_{ij} = \begin{cases} 1 & \text{if words } i \text{ and } j \text{ are synonyms} \\ 0 & \text{otherwise} \end{cases}.$$

and

$$T^{ant}_{ij} = \begin{cases} -1 & \text{if words } i \text{ and } j \text{ are antonyms} \\ 0 & \text{otherwise} \end{cases}.$$

T^{syn} is the synonym graph and T^{ant} is the antonym graph. The signed graph can then be written in matrix form as $\hat{W} = \gamma W + \beta^{ant} T^{ant} \odot W + \beta^{syn} T^{syn} \odot W$, where $\odot$ computes Hadamard product (element-wise multiplication).

The parameters γ, β^{syn}, and β^{ant} are tuned to the data target dataset using cross validation. The reader should note that σ and ϵ are not found using a target dataset, but instead using cross validation and grid search to minimize the number of negative edges within clusters and the number of disconnected components in the cluster.

4 Evaluation Metrics

We evaluated the clusters using both intrinsic and extrinsic methods. For intrinsic evaluation, we used thesaurus information for two novel metrics: 1) the number of negative edges (NNE) within the clusters, which in our semantic clusters is the number of antonyms in the same cluster, and 2) the number of disconnected components (NDC) in the synonym graph, so the number of groups of words that are not connected by a synonym relation in the thesaurus. The NDC thus has the disadvantage that it is a function of the thesaurus coverage. Our third intrinsic measure uses a gold standard designed to measure how well we capture word similarity: Semantically similar words should be in the same cluster and semantically dissimilar words should not. For extrinsic evaluation, as descibed below, we measure how much our clusters help to identify text polarity. We also compare multiple word embeddings and thesauri to demonstrate the stability of our method.

5 Experiments with Synthetic Data

In order to evaluate our signed graph clustering method, we first focused on intrinsic measures of cluster quality in synthetic data. To do so, we created random signed graphs with the same proportion of positive and negative edges as in our real dataset. Figure 2 demonstrates that the number of negative edges within a cluster is minimized using our clustering algorithm on simulated data. As the number of clusters becomes large, the number of disconnected components, which includes clusters of size one, consistently increases. Determining the optimal cluster size and similarity parameters requires making a trade off between NDC and NNE. For example, in figure 2 the optimal cluster size is 20. One can see that as the number of clusters increases NNE goes to zero, but the number of disconnected components becomes the number of vertices. In the extreme case all clusters contain one vertex. K-means, also shown in figure 2, does not optimize NNE.

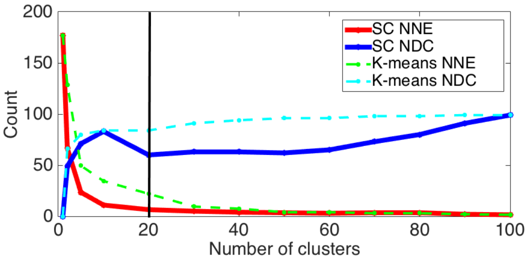

Figure 2: The relation between disconnected component (NDC) and negative edge (NNE) using simulated signed graphs with 100 vertices.

6 Experimental Setup

6.1 Word Embeddings

We used four different word embedding methods for evaluation: Skip-gram vectors (word2vec) (Mikolov et al., 2013), Global vectors (GloVe) (Pennington et al., 2014), Eigenwords (Dhillon et al., 2015), and Global Context (GloCon) (Huang et al., 2012); however, we only report the results for word2vec, which is the most popular word embedding (see the supplemental material for other embeddings). We used word2vec 300 dimensional embeddings which were trained on several billion words of English: the Gigaword and the English discussion forum data gathered as part of BOLT. Tokenization was performed using CMU's Twokenize.[4]

6.2 Thesauri

Several thesauri were used in order to test the robustness including Roget's Thesaurus (Roget, 1852), the Microsoft Word English (MS Word) thesaurus from Samsonovic et al. (2010) and WordNet 3.0 (Miller, 1995).

We chose a subset of 5108 words for the training dataset, which had high overlap between various sources. Changes to the training dataset had minimal effects on the optimal parameters. Within the training dataset, each of the thesauri had roughly 3700 antonym pairs; combined they had 6680. However, the number of distinct connected components varied, with Roget's Thesaurus having the fewest (629), and MS Word Thesaurus (1162) and WordNet (2449) having the most. These ratios were consistent across the full dataset.

6.3 Gold Standard SimLex-999 And SimVerb-3500

Following the analysis of Vlachos et al. (2009), we threshold the semantically similar datasets to find word pairs which should or should not belong to the same cluster. As ground truth, we extracted 120 semantically similar words from SimLex-999 with a similarity score greater than 8 out of 10. SimLex-999 is a gold standard resource for semantic similarity, not relatedness, based on ratings by human annotators.

Our 120 pair subset of SimLex-999 has multiple parts-of-speech including Noun-Noun pairs, Verb-Verb pairs and Adjective-Adjective pairs. Within SimVerb-3500, we used a subset of 318 semantically similar verb pairs.

The community is attempting to define better gold standards; however, currently these are the best datasets that we are aware of. We tried to use WordNet, Roget, and the Paraphrase Database (PPDB) (Ganitkevitch et al., 2013) as a gold standard, but manual inspection as well as empirical results showed that none of the automatically generated datasets were a sufficient gold standard. Possibly the symmetric pattern of (Schwartz et al., 2015) would have been sufficient; we did not have time to validate this.

6.4 Stanford Sentiment Treebank

We also evaluated our clusters by using them as features for predicting sentiment, using sentiment treebank [5] (Socher et al., 2013) with coarse-grained labels on phrases and sentences from movie review excerpts. This dataset is widely used for the evaluation of sentiment analysis. We used the standard partition of the treebank into training (6920), development (872), and test (1821) sets.

7 Cluster Evaluation

Table 1 shows the four most-associated words with "accept" using different methods.

We now turn to quantitative measures of word similarity and synonym cluster quality.

7.1 Comparison with K-means and Normalized Cuts

In order to assess the model we tested (1) K-means, (2) normalized cuts without thesaurus, and (3) signed normalized cuts. As a baseline, we created clusters using K-means on the original word2vec vector representations where the number of K clusters was set to 750.

Table 2 shows the relative ratios of the different clustering methods of with respect to antonym pair inclusion and the number of disconnected components within the clusters. For both methods, over twenty percent of the clusters contain antonym pairs even though the median cluster size is six. Signed clustering radically reduced the number of antonyms within clusters compared to the other methods.

[4] https://github.com/brendano/ark-tweet-nlp

[5] http://nlp.stanford.edu/sentiment/treebank.html

Ref word	Roget	WordNet	MS Word	W2V	SC W2V
accept	adopt accept your fate be fooled by acquiesce	agree get fancy hold	take swallow consent assume	accepts reject agree accepting	grant permit let okay

Table 1: Qualitative comparison of clusters.

Method	Antonym Ratio	DC Ratio
K-Means	0.24	0.95
NC	0.21	0.97
SC	0.06	0.49

Table 2: Clustering evaluation of K-means, normalized cuts, and signed normalized cuts with 750 clusters. Ratio of clusters with containing one or more antonym pair and ratio of clusters with disconnected components.

Method	Acc SimLex	Err
MSW Lookup	0.70	0
Roget Lookup	0.63	0
WordNet Lookup	0.43	0
Combined Lookup	0.90	0
NC(W2V)	0.36	0.05
SC (W2V)	**0.67**	0
Lookup + NC(W2V)	0.91	0.05
Lookup + **SC**(W2V)	**0.96**	0
MSW + **SC**(W2V)	0.95	0

Table 3: Clustering evaluation using SimLex-999 with 120 word pairs having similarity score over 8. SC stands for our signed clustering and NC is standard normalized cuts. **SC**(W2V) are the word clusters from signed clustering using word2vec and the combined thesauri. **Err** is the proportion of dissimilar words (with score < 2) present in the same cluster.

8 Empirical Results

Tables 3 and 5 present our main result. When using our signed clustering method with similar words, as labeled by SimLex-999 and SimVerb-3500, our clustering accuracy increased by 5% on both SimLex-999 and SimVerb-3000. Furthermore, by combining the thesauri lookup with our clustering, we achieved almost perfect accuracy (96%). Table 5 shows the sentiment analysis task performance. Our method outperforms all methods with similar complexity; however, we did not reach state-of-the-art results when compared to much more complex models which also use a richer dataset.

8.1 Evaluation Using Word Similarity Datasets

In a perfect setting, all word pairs rated highly similar by human annotators would be in the same cluster, and all words which were rated dissimilar would be in different clusters. Since our clustering algorithm produced sets of words, we used this evaluation instead of the more commonly reported correlations.

In table 3 we show the results of the evaluation with SimLex-999. Combining thesaurus lookup and word2vec+CombThes clusters, labeled as Lookup + **SC**(W2V), yielded an accuracy of 0.96 (5 errors). Note that clusters using word2vec with normalized cuts does not improve accuracy. The MSW thesaurus has much lower coverage, but 100 % accuracy, which is why when combined with the signed clustering the performance is 0.95. In table 3 we state the proportion of clusters containing dissimilar words as a sanity check for cluster size. (See supplemental material for full cluster size optimization information.) Another important result is that the verb accuracy yielded the largest accuracy gains, consistent with the results of Schwartz et al. (2015).

Table 4 clearly shows that the overall performance of all methods is lower for verb similarity. However, the improvement using both signed clustering as well as thesaurus look is also larger.

8.2 Sentiment Analysis

We trained an l_2-norm regularized logistic regression (Friedman et al., 2001) and simultaneously γ, β^{syn}, and β^{ant} using our word clusters in order to predict the coarse-grained sentiment at the sentence level. The γ and β parameters were found using a portion of the data where we iteratively switch between the logistic regression and the parameters, holding each fixed. However, hyperparameters σ and ϵ, and the number of clusters

Method	Acc SimVerb
MSW Lookup	0.45
Roget Lookup	0.59
WordNet Lookup	0.43
Combined Lookup	0.83
NC(W2V)	0.24
SC (W2V)	**0.56**
Lookup + NC(W2V)	0.83
Lookup + **SC**(W2V)	**0.88**

Table 4: Clustering evaluation using SimVerb-3500 with 317 word pairs having similarity score over 8. SC stands for our signed clustering and NC is standard normalized cuts. **SC**(W2V) are the word clusters from signed clustering using word2vec and the combined thesauri.

Model	Accuracy
NB (Socher et al., 2013)	0.818
VecAvg (W2V) (Faruqui et al., 2015)	0.812
RVecAvg (W2V) (Faruqui et al., 2015)	0.821
RNN(Socher et al., 2013)	0.824
NC(W2V)	0.79
SC(Thes)	0.752
SC(W2V)	**0.836**

Table 5: Sentiment analysis accuracy for binary predictions of signed clustering algorithm (SC) versus other models. **SC**(W2V) are the signed clusters using word2vec word representations.

K were optimized minimizing error using grid search. We compared our model against existing models: Naive Bayes with bag of words (NB) (Socher et al., 2013), sentence word embedding averages (VecAvg), retrofitted sentence word embeddings (RVecAvg) (Faruqui et al., 2015) that incorporate thesaurus information, simple recurrent neural networks (RNN), and two baselines of normalized cuts and signed normalized cuts using only thesaurus information.

While the state-of-the art Convolutional Neural Network (CNN) (Kim, 2014) is at 0.881, our model performs quite well with much less information and complexity. Table 5 shows that signed clustering outperforms the baselines of Naive Bayes, normalized cuts, and signed cuts using just thesaurus information. Furthermore, we outperform comparable models, including retrofitting, which has thesaurus information, and the recurrent neural network, which has access to domain specific context information.

Signed clustering using only thesaurus information (**SC(Thes)**) performed significantly worse than all other methods. This was largely due to low coverage; rare words such as “WOW” and “???” are not covered. As expected, because normalized cut clusters include antonyms, the method performs worse than others. Nonetheless the improvement from 0.79 to 0.836 is quite drastic.

9 Conclusion

We developed a novel theory for signed normalized cuts and an algorithm for finding their discrete solution. We showed that we can find superior semantically similar clusters which do not require new word embeddings but simply overlay thesaurus information on preexisting ones. The clusters are general and can be used with many out-of-the-box word embeddings. By accounting for antonym relationships, our algorithm greatly outperforms simple normalized cuts. Finally, we examined our clustering method on the sentiment analysis task from Socher et al. (2013) sentiment treebank dataset and showed that it improved performance versus comparable models.

Our automatically generated clusters give better coverage than manually constructed thesauri. Our signed spectral clustering method allows us to incorporate the knowledge contained in these thesauri without modifying the word embeddings themselves. We further showed that use of the thesauri can be tuned to the task at hand.

Our signed spectral clustering method could be applied to a broad range of NLP tasks, such as prediction of social group clustering, identification of personal versus non-personal verbs, and analyses of clusters which capture positive, negative, and objective emotional content. It could also be used to explore multi-view relationships, such as aligning synonym clusters across multiple languages. Another possibility is to use thesauri and word vector representations together with word sense disambiguation to generate semantically similar clusters for multiple senses of words. Furthermore, signed spectral clustering has broader applications such as cellular biology, social networking, and electricity networks. Finally, we plan to extend the hard signed clustering presented here to probabilistic soft clustering.

References

Mikhail Belkin and Partha Niyogi. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation* 15(6):1373–1396.

Dorwin Cartwright and Frank Harary. 1956. Structural balance: a generalization of heider's theory. *Psychological review* 63(5):277.

Kai-Wei Chang, Wen-tau Yih, and Christopher Meek. 2013. Multi-relational latent semantic analysis. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1602–1612. http://aclweb.org/anthology/D13-1167.

Kai-Yang Chiang, Cho-Jui Hsieh, Nagarajan Natarajan, Inderjit S. Dhillon, and Ambuj Tewari. 2014. Prediction and clustering in signed networks: A local to global perspective. *Journal of Machine Learning Research* 15:1177–1213. http://jmlr.org/papers/v15/chiang14a.html.

James Richard Curran. 2004. From distributional to semantic similarity .

Paramveer S. Dhillon, Dean P. Foster, and Lyle H. Ungar. 2015. Eigenwords: Spectral word embeddings. *Journal of Machine Learning Research* 16:3035–3078. http://jmlr.org/papers/v16/dhillon15a.html.

Manaal Faruqui, Jesse Dodge, Kumar Sujay Jauhar, Chris Dyer, Eduard Hovy, and A. Noah Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 1606–1615. https://doi.org/10.3115/v1/N15-1184.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

Jean Gallier. 2016. Spectral theory of unsigned and signed graphs applications to graph clustering: a survey. *arXiv preprint arXiv:1601.04692* .

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 758–764. http://aclweb.org/anthology/N13-1092.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simverb-3500: A large-scale evaluation set of verb similarity. *arXiv preprint arXiv:1608.00869* .

Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review* 53(2):217–288.

Frank Harary. 1953. On the notion of balance of a signed graph. *The Michigan Mathematical Journal* 2(2):143–146.

Zellig S Harris. 1954. Distributional structure. *Word* .

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *arXiv preprint arXiv:1408.3456* .

Yao Ping Hou. 2005. Bounds for the least laplacian eigenvalue of a signed graph. *Acta Mathematica Sinica* 21(4):955–960.

Eric Huang, Richard Socher, Christopher Manning, and Andrew Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 873–882. http://aclweb.org/anthology/P12-1092.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pages 1746–1751. https://doi.org/10.3115/v1/D14-1181.

Ravikrishna Kolluri, Jonathan Richard Shewchuk, and James F O'Brien. 2004. Spectral surface reconstruction from noisy point clouds. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. ACM, pages 11–21.

Jérôme Kunegis, Stephan Schmidt, Andreas Lommatzsch, Jürgen Lerner, Ernesto William De Luca, and Sahin Albayrak. 2010. Spectral analysis of signed graphs for clustering, prediction and visualization. In *SDM*. SIAM, volume 10, pages 559–559.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*. http://aclweb.org/anthology/P98-2127.

Pedro Mercado, Francesco Tudisco, and Matthias Hein. 2016. Clustering signed networks with the geometric mean of laplacians. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*. Curran Associates, Inc., pages 4421–4429. http://papers.nips.cc/paper/6164-clustering-signed-networks-with-the-geometric-mean-of-laplacians.pdf.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

M. Saif Mohammad, J. Bonnie Dorr, Graeme Hirst, and D. Peter Turney. 2013. Computing lexical contrast. *Computational Linguistics* 39(3). https://doi.org/10.1162/COLI_a_00143.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, M. Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 142–148. https://doi.org/10.18653/v1/N16-1018.

Masataka Ono, Makoto Miwa, and Yutaka Sasaki. 2015. Word embedding-based antonym detection using thesauri and distributional information. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 984–989. https://doi.org/10.3115/v1/N15-1100.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pages 1532–1543. https://doi.org/10.3115/v1/D14-1162.

The Nghia Pham, Angeliki Lazaridou, and Marco Baroni. 2015. A multitask objective to inject lexical contrast into distributional semantics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics, pages 21–26. https://doi.org/10.3115/v1/P15-2004.

Syama Sundar Rangapuram and Matthias Hein. 2012. Constrained 1-spectral clustering. *International conference on Artificial Intelligence and Statistics (AISTATS)* 22:1143—1151.

Delip Rao and Deepak Ravichandran. 2009. Semi-supervised polarity lexicon induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*. Association for Computational Linguistics, pages 675–682. http://aclweb.org/anthology/E09-1077.

Peter Mark Roget. 1852. *Roget's Thesaurus of English Words and Phrases...*. Longman Group Ltd.

Alexei V Samsonovic, Giorgio A Ascoli, and Jeffrey Krichmar. 2010. Principal semantic components of language and the measurement of meaning. *PloS one* 5(6):e10921.

Silke Scheible, Sabine Schulte im Walde, and Sylvia Springorum. 2013. Uncovering distributional differences between synonyms and antonyms in a word space model. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*. Asian Federation of Natural Language Processing, pages 489–497. http://aclweb.org/anthology/I13-1056.

Roy Schwartz, Roi Reichart, and Ari Rappoport. 2015. Symmetric pattern based word embeddings for improved word similarity prediction. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, pages 258–267. https://doi.org/10.18653/v1/K15-1026.

Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22(8):888–905.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, D. Christopher Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1631–1642. http://aclweb.org/anthology/D13-1170.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 1555–1565. https://doi.org/10.3115/v1/P14-1146.

Peter D Turney, Patrick Pantel, et al. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research* 37(1):141–188.

Andreas Vlachos, Anna Korhonen, and Zoubin Ghahramani. 2009. *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, Association for Computational Linguistics, chapter Unsupervised and Constrained Dirichlet Process Mixture Models for Verb Clustering, pages 74–82. http://aclweb.org/anthology/W09-0210.

Jin Wang, Liang-Chih Yu, K Robert Lai, and Xuejie Zhang. 2016. Community-based weighted graph model for valence-arousal prediction of affective words. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24(11):1957–1968.

Wen-tau Yih, Geoffrey Zweig, and John Platt. 2012. Polarity inducing latent semantic analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, pages 1212–1222. http://aclweb.org/anthology/D12-1111.

Stella X Yu and Jianbo Shi. 2003. Multiclass spectral clustering. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, pages 313–319.

Ron Zass and Amnon Shashua. 2005. A unifying approach to hard and probabilistic clustering. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. IEEE, volume 1, pages 294–301.

Jingwei Zhang, Jeremy Salwen, Michael Glass, and Alfio Gliozzo. 2014. Word semantic representations using bayesian probabilistic tensor factorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pages 1522–1531. https://doi.org/10.3115/v1/D14-1161.