Leveraging Sentiment to Compute Word Similarity

Abstract

In this paper, we introduce a new WordNet based similarity metric, SenSim, which incorporates sentiment content (*i.e.*, degree of positive or negative sentiment) of the words being compared to measure the similarity. The proposed metric is based on the hypothesis that knowing the sentiment is beneficial in measuring the similarity. To verify this hypothesis, we measure and compare the annotator agreement for 2 annotation strategies: 1) sentiment information of a pair of words is considered while annotating and 2) sentiment information of a pair of words is not considered while annotating. Inter-annotator correlation scores show that the agreement is better when the two annotators consider sentiment information while assigning a similarity score to a pair of words.

We use this hypothesis to measure the similarity between a pair of words. Specifically, we represent each word as a vector containing the sentiment scores of all the content words in the WordNet gloss of the words. These sentiment scores are derived from a sentiment lexicon. We then measure the cosine similarity between the two vectors. We perform both intrinsic and extrinsic evaluation of SenSim. As a part of intrinsic evaluation, we calculate the correlation score with gold standard data and compare it with other popular WordNet based metrics. We find that SenSim has better correlation than other similarity metrics. Further, as a part of extrinsic evaluation, we use SenSim in an application. We evaluate SenSim for *mitigating unknown feature problem in supervised sentiment classification using replacement strategy based on similarity metrics* as proposed by Balamurali et al. (2011). Our results show that new metric performs better than all the existing metrics used for comparison.

1 Introduction

Use of similarity metrics is unavoidable in many Natural Language Processing (NLP) systems. They form core of many NLP tasks like Word Sense disambiguation (Banerjee & Pedersen 2002), malapropism detection (Hirst & St-Onge 1997), context sensitive spelling correction (Patwardhan 2003) *etc*. The underlying principle of these metrics has been distributional similarity in terms of their meaning. For example, *refuge* and *asylum* are similar words because they have the same meaning and similar set of words accompany them in a given context. Based on the meaning alone, these words are mutually replaceable.

At present, there are various advanced text editors which have the ability to replace a word based on the meaning suitable for the domain/genre in which article is being written. To select an appropriate replacement word, they follow a similarity based on meaning alone. Motivated by the idea of sublanguages (Grishman 2001), we believe similarity based on meaning alone cannot suffice this need. For example, in the previous case, even though *refuge* can be replaced with *asylum*, *mad house* cannot be used to do so. This is because *mad house* evokes a negative connotation or sentiment which makes the word unsuitable for replacement.

In this paper, we propose a new WordNet based similarity metric, SenSim, which takes into consideration the sentiment content (i.e., degree of positive or negative sentiment) of words being compared. We create vector representations of WordNet glosses and compare their cosines to calculate the similarity score. To include the sentiment content of the words being compared, we include sentiment scores of the content words of the gloss into the vector. The main contribution of this paper is in addressing the following question

Does inclusion of sentiment content as an additional parameter for comparison improve similarity measurement?

We perform an intrinsic and extrinsic evaluation of the SenSim metric. As a part of intrinsic evalua-

tion, we manually annotate a set of 48 random word pairs based on their word similarity on a scale of 1-5 and calculate the correlation between our metric and annotator scores. We also calculate the correlation between three popular WordNet based similarity metrics and the annotated dataset (gold standard). Our results show that the new metric has a better correlation with the annotator scores than the other metrics used for this study.

For extrinsic evaluation, we compare the effect of SenSim metric *to mitigate the effect of unknown feature problem in supervised sentiment classification using synset replacement strategy* as proposed by Balamurali et al. (2011). Under this application of similarity metrics, authors propose to replace features not present in test set with *similar* features present in the training set using a similarity metric. Our results show that SenSim based document level sentiment classifier performs better than other WordNet based metrics.

The rest of the paper is organized as follows: Section 2 describes the related work and distinguishes our work from the existing similarity metrics. We describe our metric and related terminologies in Section 3. We explain evaluation strategy in section 4. Section 5 describes existing similarity metrics that we use for comparison. Experimental setup is given in Section 6. Results of the experiments are discussed and analyzed in Section 7. Section 8 concludes the paper.

2 Related work

Various approaches for evaluating the similarity between two words can be broadly classified into two categories: edge-based methods and information-content-based methods. One of the earliest works in edge-based calculation of similarity is by Rada et al. (1989), where in, they propose a metric "Distance" over a semantic net of hierarchical relations as the shortest path length between the two nodes. This has been the basis for all the metrics involving simple edge-counting to calculate the distance between two nodes. However, the simple edge-counting fails to consider the variable density of nodes across the taxonomy. It also fails to include relationships other than the is-a relationship, thus, missing out on important information in a generic semantic ontology, like WordNet.

In contrast to edge-based methods, Richardson et al. (1994) and Resnik (1995*a*) propose a node-based approach to find the semantic similarity. They approximate conceptual similarity between two WordNet concepts as the maximum information content among classes that subsume both the concepts. Resnik (1995*b*) advanced this idea by defining the information content of a concept based on the probability of encountering an instance of that concept. Alternatively, Wu & Palmer (1994) compare two concepts based on the length of the path between the root of the hierarchy and the least common subsumer of the concepts.

Jiang & Conrath (1997) and Leacock et al. (1998) combine the above two approaches by using the information content as weights for the edges between concepts. They further reinforce the definition of information content of a concept by adding corpus statistical information.

Instead of measuring the similarity of concepts, some other approaches measure their relatedness. Hirst & St-Onge (1997) introduce an additional notion of direction along with the length of paths for measuring the relatedness of two concepts. Banerjee & Pedersen (2003) and Patwardhan (2003) leverage the gloss information present in WordNet in order to calculate the relatedness of two concepts. Banerjee & Pedersen (2003) assigns relatedness scores based on the overlap between the gloss of the two concepts. Patwardhan (2003) use a vector representation of the gloss, based on the context vector of the terms in the gloss. The relatedness is then the cosine between the gloss vectors of the two concepts.

Our work is most related to the work of Wan & Angryk (2007) which improves on Banerjee & Pedersen (2003) and Patwardhan (2003) by including relations other than the is-a relationship. They use an extended gloss definition for a concept which is defined as the original gloss appended by the gloss of all the concepts related to the given concept. They create concept vectors for each sense based on which they create context vectors which are an order higher to the concept vectors. Finally, they use cosine of the angle between the vectors of the different concepts to find their relatedness. This approach is better than other approaches as it captures the context of the concepts to a much larger extent. However, all

these methods lack on a common ground. They fail to incorporate sentiment information in calculating the similarity/relatedness of two concepts. We postulate that sentiment information is crucial in finding the similarity between two concepts.

3 SenSim metric

The underlying hypothesis that we follow for creating this metric is that *knowing the sentiment is beneficial in measuring the similarity*. In order to implement a metric based on this hypothesis, we incorporate sentiment values of the words being compared. Similar to Wan & Angryk (2007), we follow WordNet gloss based comparison technique to develop this metric. Gloss based technique has an inherent advantage over edge-based and information-content-based metrics as it is applicable to all POS categories without any distinction.

3.1 Gloss vector

We represent gloss of a synset in the form of a vector, we define this vector as gloss vector. To obtain the gloss of the words being compared, the corresponding sense used for each word needs to be known. We assume that we are provided with the synset corresponding to each word that needs to be compared. In other scenarios where the synset corresponding to word are not given, a close approximation can be taken by using respective WordNet first sense. Each dimension of the gloss vector represents a sentiment score of the respective content word. To obtain the sentiment scores, we use an external sentiment lexicon and assign sentiment values based on different scoring functions.

We use SentiWordNet 1.0[1] as the external sentiment lexicon to incorporate the sentiment values in the gloss vector. This Wordnet based resource has polarity scores attached to synsets (Esuli & Sebastiani 2006). Each synset in this resource is marked with 3 scores: a positive score, a negative score and an objective score, with the scores summing up to 1. As the sentiment scores are attached to synsets rather than lexemes, we disambiguate the WordNet gloss to obtain the corresponding synsets. Based on synsets thus found, we assign sentiment scores to each dimension of the gloss vector.

[1] http://sentiWordNet.isti.cnr.it/

Representing gloss in the form of vectors is not new, but novelty of our approach is in the incorporation of sentiment score to each dimension of the gloss vector.

3.2 Augmenting gloss vector

Gloss contains few content words averaging between 5-7. This creates a sparse vector space. To reduce the sparseness of the gloss vector, we augment the original gloss with the gloss of the *related synsets*. We use different WordNet relations , based on the POS category, to find the related synsets. Apart from the *adjacent* related synset, we add more context by further expanding the related synsets using synsets of content words of the original gloss.

Not all WordNet relations can be used for the expansion procedure as degree of sentiment content may change or not get carried to the next level. By taking relative transfer of the sentiment content from one synset to another for different WordNet relations, we empirically found a set of WordNet relations suitable for each POS category. Details of the WordNet relations used for expansion process are given in Table 1.

POS	WordNet relations used for expansion
Nouns	*hypernym, hyponym, nominalization*
Verbs	*nominalization, hypernym, hyponym*
Adjectives	*also see, nominalization, attribute*
Adverbs	*derived*

Table 1: WordNet relations used for enhancing the context of Gloss vector

3.3 Scoring function

As SentiWordNet provides 3 scores for each synset: positive score, negative score and objective score, we devise a scoring function to capture the sentiment content of the words as a single real value. We explain four variants of the scoring function used in this study below:

Sentiment Difference (SD)

Difference between the positive score and the negative score is taken as the sentiment score of the synset concerned.

$$Score_{SD}(A) = SWN_{pos}(A) - SWN_{neg}(A)$$

Here $SWN_{pos}(A)$ signifies the positive score pertaining to the synset A. The sign of the score represents the orientation of the sentiment. If the sign is negative, the word has a negative connotation otherwise it has a positive connotation. If the value is zero, it is objective in nature.

Sentiment Max (SM)

For this function, we use the greater of the positive or negative score of the synset as sentiment score of the synset concerned.

$$Score_{SM}(A) = max(SWN_{pos}(A), SWN_{neg}(A))$$

The orientation of the word is again distinguished by the sign. For negative connotation, a negative of the score returned is used else a positive score is taken.

Sentiment Threshold Difference (TD)

As the gloss is represented in the form of a vector with each dimension representing a sentiment score, dimensions which have zero magnitude may not contribute to sentiment content but the presence of each dimension is necessary for overall similarity computation. In order to avoid such scenarios, we introduce a threshold value that ensures, in case of objective words, a zero value is never encountered. We take a threshold value of 1 to compute the variant of SD scoring function.

$$Score_{TD}(A) = sign(SWN_{pos}(A) - SWN_{neg}(A)) * (1 + abs(SWN_{pos}(A) - SWN_{neg}(A)))$$

Sentiment Threshold Max (TM)

SM scoring function is modified to handle zero magnitude problem as explained above.

$$Score_{TM}(A) = sign(max(SWN_{pos}(A), SWN_{neg}(A))) * (1 + abs(max(SWN_{pos}(A), SWN_{neg}(A)))$$

3.4 Computing similarity

To compute the similarity between two word pairs, we take the cosine similarity of their corresponding gloss vectors.

$SenSim_x$(A,B) = $cosine(gloss_{vec}$(sense(A), $gloss_{vec}$(sense(B))

where

$gloss_{vec}$ = 1:$score_x$(1) 2:$score_x$(2) ..... n:$score_x$(n)

$score_x$(Y) = Sentiment score of word Y using scoring function x

x = Scoring function of type $SD/SM/TD/TM$

4 Evaluation

To evaluate SenSim, we follow two methodologies: an intrinsic evaluation and an extrinsic evaluation. For intrinsic evaluation, we compare the correlation of the metric with a gold standard dataset. We also compare correlation of different existing similarity metrics with this gold standard and show how our new metric performs. To perform an extrinsic evaluation, we use SenSim metric to mitigate the effect of the unknown feature problem in supervised sentiment classification. Details of the same are explained in the following subsections.

4.1 Intrinsic evaluation: correlation with human annotators

We develop a new similarity dataset and manually annotate them with similarity scores. We did not use existing similarity datasets as we require the correct sense of the words being compared.

Dataset for annotation task

We chose 48 random word pairs for this task with each POS category having 12 word pairs each. Sentences, containing these words, are constructed to get the exact sense of these word pairs. Based on these sentences, words are sense disambiguated using WordNet 2.1 to get the corresponding synsets. A part of the word pairs used in this paper are given in Table2 [2]. Each word pair has a WordNet synset offset, prefixed with the POS, category attached to it.

Annotation strategy

To test our hypothesis that *incorporation of sentiment into a similarity metric gives better result than comparing similarity based on meaning alone*, we perform a similarity annotation task between two

[2]The complete set of word pairs is not included due to lack of space but is available on request

Word Pairs	
regular_42374008	*accustomed_426235*
tardily_3100974	*lately_3108293*
randomly_371128	*specifically_341621*
pretentious_41915502	*arrogant_41957189*
defense_1811665	*attack_1958708*

Table 2: A section of dataset used for experiments

human annotators. The annotators were asked to annotate word pairs using two different strategies.

1. *Annotation based on Meaning* : Instruction were given to each annotator to give a score between 1-5 to word pairs based on semantic similarity with a score of 5 representing synonymous word pairs and 1 representing no relation between the words.

2. *Annotation based on Sentiment and Meaning combined* : In this case, the annotators were asked to rate on a scale between 1-5 whether words were interchangeable given similar context and similar sentiment content. A score of 5 implies perfect interchangeability.

The dataset generated thus forms our gold standard data. Correlation between the annotator score is taken to test the hypothesis. As a part of this evaluation, we also take the correlation scores between different existing similarity metrics with the gold standard data, and compare the respective correlation scores with SenSim.

4.2 Extrinsic evaluation: synset replacement using similarity metrics

In this section, we evaluate SenSim metric using an application of similarity metrics, and compare the application performance with that of widely used similarity metrics. We use the synset replacement strategy using similarity metrics by Balamurali et al. (2011) for evaluating our metric. In this study, the authors showed that similarity metrics can be used to mitigate the effect of unseen feature problem in supervised classification. The objective of their study is to classify documents based on their sentiment content into positive class or negative class. Each document to be classified is represented as a group of synsets obtained after sense disambiguation of the content words of the document. A document level sentiment classifier is created based on the training corpus and its performance measured is on the test corpus. A trained classifier will not perform with same accuracy if new features are found in the test corpus. To mitigate this effect, they follow a *replacement strategy*. In this strategy, if a synset encountered in a test document is not found in the training corpus, it is replaced by one of the synsets present in the training corpus. The substitute synset is determined on the basis of its similarity with the concerned synset in the test document. The synset that is replaced is referred to as an *unseen synset* as it is not known to the trained model.

```
Input: Training Corpus, Test Corpus,
Similarity Metric
Output: New Test Corpus
T:= Training Corpus;
X:= Test Corpus;
S:= Similarity metric;
train_concept_list = get_list_concept(T) ;
test_concept_list = get_list_concept(X);
for each concept C in test_concept_list do
    temp_max_similarity = 0 ;
    temp_concept = C ;
    for each concept D in train_concept_list do
        similarity_value = get_similarity_value(C,D,S);
        if (similarity_value > temp_max_similarity) then
            temp_max_similarity= similarity_value;
            temp_concept = D ;
        end
    end
    C = temp_concept ;
    replace_synset_corpus(C,X);
end
Return X ;
```

Algorithm 1: Synset replacement using similarity metric

Algorithm 1 shows the replacement algorithm devised by Balamurali et al. (2011). The algorithm follows from the fact that the similarity value for a synset with itself is maximum. Similarity metrics used in this study are explained in the following section.

5 Metrics used for comparison

We compare SenSim with three existing similarity metric. They are:

LIN: The metric by Lin (1998) uses the infor-

Annotation Strategy	Overall	NOUN	VERB	ADJECTIVES	ADVERBS
Meaning	0.768	0.803	0.750	0.527	0.759
Meaning + Sentiment	0.799	0.750	0.889	0.720	0.844

Table 3: Pearson correlation coefficient between two annotators for various annotation strategy

mation content individually possessed by two concepts in addition to that shared by them. The information content shared by two concepts A and B is given by their most specific subsumer (lowest superordinate(*lso*)). Thus, this metric defines the similarity between two concepts as

$$sim_{LIN}(A,B) = \frac{2 \times \log Pr(lso(A,B))}{\log Pr(A) + \log Pr(B)}$$

Lesk: Each concept in WordNet is defined through gloss. To compute the Lesk similarity (Banerjee & Pedersen 2002) between A and B, a scoring function based on the overlap of words in their individual glosses is used.

Leacock and Chodorow (LCH): To measure similarity between two concepts A and B, Leacock & Chodorow (1998) compute the shortest path through hypernymy relation between them under the constraint that there exists such a path. The final value is computed by scaling the path length by the overall taxonomy depth (D).

$$sim_{LCH}(A,B) = -\log\left(\frac{len(A,B)}{2D}\right)$$

6 Experimental setup

Word sense disambiguation of WordNet glosses is carried out using the WSD engine by Zhong & Ng (2010). It is an all-words generic WSD engine with an accuracy of 82% on standard WSD corpus. We use WordNet::Similarity 2.05 package by Pedersen et al. (2004) for computing the similarity by other metric scores mentioned in this paper. We use Pearson correlation coefficient to find the inter-annotator agreement.

For evaluation based on synset replacement using similarity metrics, we use the dataset provided by Balamurali et al. (2011). The experiments are performed using C-SVM (linear kernel with default parameters[3]), using bag-of-synsets as features, available as a part of LibSVM[4] package. All classification results reported are average of five-fold cross-validation accuracies.

To evaluate the result, we use accuracy, recall and precision as the metrics. Classification accuracy defines the ratio of the number of true instances to the total number of instances. Recall is calculated as a ratio of the true instances found to the total number of false positives and true positives. Precision is defined as the number of true instances divided by the number of true positives and false negatives. Positive Precision (PP) and Positive Recall (PR) are precision and recall for positive documents while Negative Precision (NP) and Negative Recall (NR) are precision and recall for negative documents.

7 Results and discussion

7.1 Sentiment as a parameter for finding similarity

Table 3 shows correlation scores between two annotators for different annotation strategies. Apart from the correlation of the complete word pairs, it also shows POSwise correlation. From the results, it is clearly evident that similarity is best captured when sentiment is also included. The annotation strategy involving a combination of meaning and sentiment has a better correlation among annotators(0.799) than the one which considers meaning alone. This verifies the hypothesis that taking sentiment content of words being compared is beneficial in assessing the similarity between them.

A POSwise analysis of Table 3 suggests that apart from the Noun category, all other categories have a better correlation among annotators in assessing the sentiment based similarity annotation strategy. This may be due to the fact that in case of word pairs belonging to the Noun category, sentiment does not

[3] C=0.0,ϵ=0.0010

[4] http://www.csie.ntu.edu.tw/ cjlin/libsvm

Metric Used	Overall	NOUN	VERB	ADJECTIVES	ADVERBS
LESK	0.22	0.51	-0.91	0.19	0.37
LIN	0.27	0.24	0.00	NA	NA
LCH	0.36	0.34	0.44	NA	NA
SenSim (SD)	0.46	0.73	0.55	0.08	0.76
SenSim (TD)	0.50	0.62	0.48	0.06	0.54
SenSim (SM)	0.45	0.73	0.55	0.08	0.59
SenSim (TM)	0.48	0.62	0.48	0.06	0.78

Table 4: Pearson Correlation(r) of various metrics with Gold standard data

play much role. The highest correlation is seen for Verbs. In case of Adjectives, annotators have a fairly high correlation. Since most of the Adjectives are sentiment bearing words, annotators might have found easier to compare them. In summary, a similarity metric which incorporates sentiments may be more beneficial to POS categories other than Nouns.

Table 4 shows the correlation between scores obtained using different similarity metrics with the scores obtained from gold standard dataset of an annotator. The correlation with respect to different POS categories is also shown. NA in some of the columns represent word pairs , belonging to those POS categories, which cannot be handled using the similarity metric concerned. For example, metrics like LIN and LCH cannot handle POS categories other than Nouns and Verbs. SenSim has a better correlation with gold standard data than other metrics. In fact, all variants of SenSim function better than the existing similarity measurement techniques used in this paper. Among different variants, SenSim using TD based scoring function performs the best. It has a correlation score of .50 whereas the nearest correlation among the other metrics is by LCH (.36). Moreover, in case of all POS categories barring Adjectives, SenSim metric has a better correlation than the rest of the metrics with gold standard data.

Although not provided in the table, reader should note that the metrics used in this study could not score all the word pairs created for this study. For example, out of 48 word pairs, SenSim could mark only 34 word pairs and among other metrics, the best count is provided by Lesk with 17 word pairs. The correlation scores shown for each metric is with respect to word pairs that had some values for comparison with the gold standard data. Thus in terms of coverage also, SenSim performs better than the metrics used in this study.

7.2 Effect of SenSim on synset replacement strategy

Table 5 shows classification result using synset replacement strategy based on similarity metrics. The results of the classifier trained on synsets alone, without any replacement, is taken as the baseline.

SenSim(TM) variant obtains the best classification accuracy among the various metrics analyzed. It achieves an accuracy of 90.17%. Even though the improvement is marginal, reader must note that no complex features are used for training this classifier. All variants of SenSim, except for SenSim(SD) achieve a 90% classification accuracy. Compared to the baseline, other WordNet metrics also have better classification accuracies.

Classification using SenSim (TM) metric has the highest positive precision. Same is the case with negative recall, it has a negative recall of 91.58%. The SenSim (SD) variant has the highest positive recall. In general, it can be seen that positive class's performance has improved by using SenSim metric for synset replacement.

8 Conclusion

In this paper, we proposed that sentiment content can aid in similarity measurement, which to date has been done on the basis of meaning alone. We verified this hypothesis by taking the correlation between annotators using different annotation strategies. Annotator correlation for the strategy involving sentiment as an additional parameter for similarity measurement was higher than the one which

Metric used	Accuracy(%)	PP	NP	PR	NR
Baseline	89.10	91.50	87.07	85.18	91.24
LSK	89.36	91.57	87.46	85.68	91.25
LIN	89.27	91.24	87.61	85.85	90.90
LCH	89.64	90.48	88.86	86.47	89.63
SenSim (SD)	89.95	91.39	88.65	87.11	90.93
SenSim (TD)	90.06	92.01	88.38	86.67	91.58
SenSim (SM)	90.11	91.68	88.69	86.97	91.23
SenSim (TM)	90.17	91.81	88.71	87.09	91.36

Table 5: Classification results of synset replacement experiment using different similarity metrics; PP-Positive Precision (%), NP-Negative Precision(%), PR-Positive Recall (%), NR-Negative Recall (%)

involved just semantic similarity. Based on this hypothesis, we introduced a new similarity metric, SenSim, which accounts for the sentiment content of the words being compared. An intrinsic evaluation of the metric with human annotated word pairs for similarity showed higher correlations than the popular WordNet based similarity metrics. We also carried out an extrinsic evaluation of SenSim on synset replacement strategy for mitigation of unknown feature problem in supervised classification. Our results suggest that apart from the overall improvement of sentiment classification accuracy, SenSim improves the classification performance of positive-class-documents .

References

Balamurali, A., Joshi, A. & Bhattacharyya, P. (2011), Harnessing wordnet senses for supervised sentiment classification, *in* 'Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing', Association for Computational Linguistics, Edinburgh, Scotland, UK., pp. 1081–1091.

Banerjee, S. & Pedersen, T. (2002), An adapted lesk algorithm for word sense disambiguation using wordnet, *in* 'Proc. of CICLing'02', London, UK, pp. 136–145.

Banerjee, S. & Pedersen, T. (2003), Extended gloss overlaps as a measure of semantic relatedness, *in* 'In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence', pp. 805–810.

Esuli, A. & Sebastiani, F. (2006), SentiWordNet: A publicly available lexical resource for opinion mining, *in* 'Proceedings of LREC-06', Genova, IT, pp. 417–422.

Grishman, R. (2001), Adaptive information extraction and sublanguage analysis, *in* 'Proceedings of the 17th International Joint Conference on Artificial Intelligence'.

Hirst, G. & St-Onge, D. (1997), 'Lexical chains as representation of context for the detection and correction malapropisms'.

Jiang, J. J. & Conrath, D. W. (1997), Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, *in* 'International Conference Research on Computational Linguistics (ROCLING X)', p. 9008.

Leacock, C. & Chodorow, M. (1998), Combining local context with wordnet similarity for word sense identification, *in* 'WordNet: A Lexical Reference System and its Application'.

Leacock, C., Miller, G. A. & Chodorow, M. (1998), 'Using corpus statistics and wordnet relations for sense identification', *Comput. Linguist.* **24**, 147–165.

Lin, D. (1998), An information-theoretic definition of similarity, *in* 'In Proceedings of ICML '98', Morgan Kaufmann, pp. 296–304.

Patwardhan, S. (2003), Incorporating dictionary and corpus information into a context vector measure of semantic relatedness, Master's thesis, University of Minnesota, Duluth.

Pedersen, T., Patwardhan, S. & Michelizzi, J. (2004), Wordnet::similarity: measuring the relat-

edness of concepts, *in* 'Demonstration Papers at HLT-NAACL'04', pp. 38–41.

Rada, R., Mili, H., Bicknell, E. & Blettner, M. (1989), 'Development and application of a metric on semantic nets', *IEEE Transactions on Systems Management and Cybernetics* **19**(1), 17–30.

Resnik, P. (1995*a*), Disambiguating noun groupings with respect to Wordnet senses, *in* D. Yarovsky & K. Church, eds, 'Proceedings of the Third Workshop on Very Large Corpora', Association for Computational Linguistics, Somerset, New Jersey, pp. 54–68.

Resnik, P. (1995*b*), Using information content to evaluate semantic similarity in a taxonomy, *in* 'In Proceedings of the 14th International Joint Conference on Artificial Intelligence', pp. 448–453.

Richardson, R., Smeaton, A. F. & Murphy, J. (1994), Using wordnet as a knowledge base for measuring semantic similarity between words, Technical report, In Proceedings of AICS Conference.

Wan, S. & Angryk, R. A. (2007), Measuring semantic similarity using wordnet-based context vectors., *in* 'SMC'07', pp. 908–913.

Wu, Z. & Palmer, M. (1994), Verb semantics and lexical selection, *in* '32nd. Annual Meeting of the Association for Computational Linguistics', New Mexico State University, Las Cruces, New Mexico, pp. 133 –138.

Zhong, Z. & Ng, H. T. (2010), It makes sense: A wide-coverage word sense disambiguation system for free text., *in* 'ACL (System Demonstrations)'10', pp. 78–83.