A Cohesion Graph Based Approach for Unsupervised Recognition of Literal and Non-literal Use of Multiword Expressions

Linlin Li and **Caroline Sporleder**
Saarland University
Postfach 15 11 50
66041 Saarbrücken
Germany
{linlin,csporled}@coli.uni-saarland.de

Abstract

We present a graph-based model for representing the lexical cohesion of a discourse. In the graph structure, vertices correspond to the content words of a text and edges connecting pairs of words encode how closely the words are related semantically. We show that such a structure can be used to distinguish literal and non-literal usages of multi-word expressions.

1 Introduction

Multiword expressions (MWEs) are defined as "idiosyncratic interpretations that cross word boundaries or spaces" (Sag et al., 2001). Such expressions are pervasive in natural language; they are estimated to be equivalent in number to simplex words in mental lexicon (Jackendoff, 1997). MWEs exhibit a number of lexical, syntactic, semantic, pragmatic and statistical idiosyncrasies: syntactic peculiarities (e.g., *by and large*, *ad hoc*), semantic non-compositionality (e.g., as in *kick the bucket* (die) and *red tape* (bureaucracy)), pragmatic idiosyncrasies (the expression is sometimes associated with a fixed pragmatic point, e.g., *good morning, good night*), variation in syntactic flexibility (e.g., *I handed in my thesis* = *I handed my thesis in* vs. *Kim kicked the bucket* ≠ **the bucket was kicked by Kim*), variation in productivity (there are various levels of productivity for different MWEs, e.g., *kick/*beat/*hit the bucket*, *call/ring/phone/*telephone up*).

These idiosyncrasies pose challenges for NLP systems, which have to recognize that an expression is an MWE to deal with it properly. Recognizing MWEs has been shown to be useful for a number of applications such as information retrieval (Lewis and Croft, 1990; Rila Mandala and Tanaka, 2000; Wacholder and Song, 2003) and POS tagging (Piao et al., 2003). It has also been shown that MWEs account for 8% of parsing errors with precision grammars (Baldwin et al., 2004). Furthermore, MWE detection is used in information extraction (Lin, 1998b) and an integral component of symbolic MT systems (Gerber and Yang, 1997; Bond and Shirai, 1997).

However, the special properties of MWEs can also be exploited to recognize MWEs automatically. There have been many studies on MWEs: identification (determining whether multiple simplex words form a MWE in a given token context, e.g. *put the sweater on* vs. *put the sweater on the table*), extraction (recognizing MWEs as word units at the type level), detecting or measuring compositionality of MWEs, semantic interpretation (interpreting the semantic association among components in MWEs).

To extract MWEs, various methods have been proposed that exploit the syntactic and lexical fixedness exhibited by MWEs, or apply various statistical measures across all co-occurrence vectors between the whole expression and its component parts (see Section 2). These methods can be used to automatically identify potentially idiomatic expressions at a type level, but they do not say anything about the idiomaticity of an expression in a particular context. While some idioms (e.g., *ad hoc*) are always used idiomatically, there are numerous others that can be used both idiomatically (see Example 1) and non-idiomatically (see Example 2).

(1) When the members of De la Guarda aren't hanging around, they're yelling and bouncing off the wall.

(2) Blinded by the sun, Erstad leaped at the wall, but the ball bounced off the wall well below his glove.

Our work aims to distinguish the literal and non-literal usages of idiomatic expressions in a

discourse context (so-called *token based classification*). It is therefore different from *type-based approaches* which aim to detect the general idiomaticity of an expression rather than its actual usage in a particular context.

We utilize the cohesive structure of a discourse (Halliday and Hasan, 1976) to distinguish literal or non-literal usage of MWEs. The basic idea is that the component words of an MWE contribute to the cohesion of the discourse in the literal case, while in the non-literal case they do not. For instance, in the literal use of *break the ice* in Example 3, the content word *ice* contributes to the overall semantic connectivity of the whole sentence by the fact that *ice* is semantically related to *water*. In contrast, in the non-literal example in 4, the word *ice* does not contribute to the overall cohesion as it is poorly connected to all the other (content) words in this specific context (*play, party, games*).

(3) The water would break the ice into floes with its accumulated energy.

(4) We played a couple of party games to break the ice.

Our approach bears similarities to Hirst and St-Onge's (1998) method for detecting malapropisms based on their non-participation in cohesive chains. However, computing such chains requires a pre-defined similarity threshold which governs whether a word is placed in a particular chain. Setting this threshold typically requires a manually labeled development set, which makes this method weakly supervised. We propose an alternative, parameter-free method in which we model the cohesive structure of a discourse as a graph structure (called *cohesion graph*), where the vertices of the graph correspond to the content words of the text and the edges encode the semantic relatedness between pairs of words. To distinguish between literal and non-literal use of MWEs, we look at how the average relatedness of the graph changes when the component words of the MWE are excluded or included in the graph (see Section 3).[1]

We first introduced the cohesion graph method in Sporleder and Li (2009). In the present paper, we provide a formalization of the graph and experiment with different vertex and edge weighting schemes. We also report on experiments with varying the size of the input context and also with pruning the graph structure automatically.

2 Related Work

Type-based MWE classification aims to extract multiword expression types in text from observations of the token distribution. It aims to pick up on word combinations which occur with comparatively high frequencies when compared to the frequencies of the individual words (Evert and Krenn, 2001; Smadja, 19993). The lexical and syntactic fixedness property can also be utilized to automatically extract MWEs (Baldwin and Villavicencio, 2002).

The study of semantic compositionality of MWEs focuses on the degree to which the semantics of the parts of an MWE contribute towards the meaning of the whole. The aim is a binary classification of the MWEs as idiosyncratically decomposable (*e.g. spill the beans*) or non-decomposable (*e.g. kick the bucket*). Several approaches have been proposed. Lin (1999) uses the substitution test[2] and mutual information (MI) to determine the compositionality of the phrase. An obvious change of the MI value of the phrase in the substitution test is taken as the evidence of the MWEs being non-compositional. Bannard et al. (2003) assume that compositional MWEs occur in similar lexical context as their component parts. The co-occurrence vector representations of verb particle construction (VPC) and the component words are utilized to determine the compositionality of the MWE.

There have also been a few token-based classification approaches, aimed at classifying individual instances of a potential idiom as literal or non-literal. Katz and Giesbrecht (2006) make use of latent semantic analysis (LSA) to explore the local linguistic context that can serve to identify multiword expressions that have non-compositional meaning. They measure the cosine vector similarity between the vectors associated with an MWE as a whole and the vectors associated with its constituent parts and interpret it as the degree to which the MWE is compositional. They report an av-

[1] By modeling lexical cohesion as a graph structure, we follow earlier approaches in information retrieval, notably by Salton and colleagues (Salton et al., 1994). The difference is that these works aim at representing similarity between larger text segments (e.g., paragraphs) in a so-called 'text' or 'paragraph relation map', whose vertices correspond to a text segment and whose edges represent the similarity between the segments (modeled as weighted term overlap).

[2] The substitution test aims to replace part of the idiom's component words with semantically similar words, and test how the co-occurrence frequency changes.

erage accuracy of 72%, but the data set used in their evaluation is small. Birke and Sarkar (2006) use literal and non-literal seed sets acquired without human supervision to perform bootstrapping learning. The new instances of potential idioms are always labeled according to the closest set. While their approach is unsupervised clustering, they do rely on some resources such as databases of idioms. Cook et al. (2007) and Fazly et al. (2009) rely crucially on the concept of *canonical form* (CForm). It is assumed that for each idiom there is a fixed form (or a small set of those) corresponding to the syntactic pattern(s) in which the idiom normally occurs. The canonical form allows for inflection variation of the heard verb but not for other variations (such as nominal inflection, choice of determiner etc.). It has been observed that if an expression is used idiomatically it typically occurs in its canonical form (Riehemann, 2001). Fazly and her colleagues exploit this behavior and propose an unsupervised method for token-based idiom classification in which an expression is classified as idiomatic if it occurs in canonical form and literal otherwise. The canonical forms are determined automatically using a statistical, frequency-based measure. They also developed statistical measures to measure the lexical and syntactic fixedness of a given expression, which is used to automatically recognize expression types, as well as their token identification in context. They report an average accuracy of 72% for their canonical form (CForm) classifier.

3 Cohesion Graph

In this section, we first give a formal definition of the cohesion graph that is used for modeling discourse connectivity, then we define the discourse connectivity. Finally, we introduced our graph-based classifier for distinguishing literal and non-literal use of MWEs.

3.1 Cohesion Graph Structure

A cohesion graph (CG) is an undirected complete graph [3] $G = (V, E)$, where

V: is a set of nodes $\{v_1, v_2, ..., v_n\}$, where each node $v_i = (t_i, id_i)$ represents a unique token in the discourse. t_i is the string form of the token, and id_i denotes the position of the token in the context.

[3] In the mathematical field of graph theory, a complete graph is a simple graph in which every pair of distinct vertices is connected by an edge. The complete graph on n vertices has $n(n-1)/2$ edges.

E: is a set of edges $\{e_{12}, e_{13}, ..., e_{(n)(n-1)}\}$, such that each edge e_{ij} connects a pair of nodes (v_i, v_j). n is the total number of tokens in the discourse that the graph models. The value of e_{ij} represents the semantic relatedness of the two tokens t_i, t_j that e_{ij} connects:

$$e_{ij} = h(t_i, t_j) \tag{5}$$

where h is a semantic relatedness assignment function. The explicit form of h will be discussed in the next section.

e_i is the average semantic relatedness of the token t_i in the discourse. It represents the average relatedness score of a certain token to its surrounding context:

$$e_i = \sum_{j=1, j\neq i}^{n} \lambda_{ij} \times e_{ij} \tag{6}$$

where λ_{ij} is the weight of the edge e_{ij}, with the constraint, $\sum_{j=1, j\neq i}^{n} \lambda_{ij} = 1$.

The edge weight function λ_{ij} allows us to weight the relatedness between two tokens, for example based on their distance in the text. The motivation for this is that the closer two tokens occur together, the more likely it is that their relatedness is not accidental. For instance, the idiom *break the ice* in Example 7 could be misclassified as literal due to there being a high relatedness score between *ice* and *snow*. The weight function is introduced so that relatedness with tokens that are closer to MWE component words counts more.

(7) The train was canceled because of the wind and **snow**. All the people in the small village train station felt upset. Suddenly, one guy broke the **ice** and proposed to play a game.

The weight function λ_{ij} is defined in terms of the inverse of the distance δ between the two token positions id_i and id_j:

$$\lambda_{ij} = \frac{\delta(id_i, id_j)}{\sum_j \delta(id_i, id_j)} \tag{8}$$

As the semantic relatedness among the MWE component words does not contain any information of how these component words are semantically involved in the context, we do not count the edges *between* the MWE component words

(as e_{45} in Figure 1). We set all the weights for connecting MWE component words to be 0, $\delta(id_i^{mwe'}, id_j^{mwe}) = 0$.

$c(G)$: is defined as the discourse connectivity of the cohesion graph. It represents the semantic relatedness score of the discourse.

$$c(G) = \sum_{i=1}^{n} (\beta_i \times e_i) \tag{9}$$

where n is the total number of tokens in the discourse, β_i is the weight of the average semantic relatedness of the token t_i with the constraint $\sum_i \beta_i = 1$. It represents the importance of the relatedness contribution of a specific token t_i in the discourse. For instance, the word *Monday* in Example 12 should be assigned less weight than the word *bilateral* as it is not part of the central theme(s) of the discourse. This is often the case for time expressions. β_i is defined as:

$$\beta_i = \frac{salience(t_i)}{\sum_j salience(t_j)} \tag{10}$$

To model the salience of a token for the semantic context of the text we use a $tf.idf$-based weighting scheme. Since we represent word tokens rather than word types in the cohesion graph, we do not need to model the term frequency tf separately, instead we set $salience$ to the log value of the inverse document frequency idf:

$$salience(t_i) = log \frac{|D|}{|\{d : t_i \in d\}|} \tag{11}$$

where D is the total number of documents in our data set and $|\{d : t_i \in d\}|$ is the number of documents in which t_i occurs. Terms which are related to the sub-topics of a document will typically only occur in a few texts in the collection, hence their idf (and often also their tf) is high and they will thus be given more weight in the graph. Terms which are not related to the central themes of a text, such as temporal expressions, will be given a lower weight. A complication arises for component words of the MWE: these occur in all of our examples and thus will receive a very low idf. This is an artifact of the data and not what we want as it means that the average connectivity of the graph virtually always increases if the MWE is excluded, causing the classifier to over-predict 'non-literal'. To counteract this effect, we set $|\{d : t_i \in d\}|$ of these words uniformly to 1.

(12) "Gujral will meet Sharif on **Monday** and discuss **bilateral** relations," the Press Trust of India added. The minister said Sharif and Gujral would be able to "break the ice" over Kashmir.

3.2 Graph-based Classifier

The cohesion graph based classifier compares the cohesion graph connectivity of the discourse including the MWE component words with the connectivity of the discourse excluding the MWE component words to check how well the MWE component words are semantically connected to the context. If the cohesion graph connectivity increases by including MWE component words, the MWE is thought to be semantically well related to its discourse. It is classified as literal (otherwise as non-literal). In other words, the cohesion graph based algorithm detects the strength of relatedness between the MWE component words and their context by calculating the discourse connectivity gain, and classifies instances as literal or non-literal based on this gain. This process is described as Formula 13 (if $\Delta c > 0$, it is literal; otherwise it is non-literal):

$$\Delta c = c(G) - c(G') \tag{13}$$

where, $c(G)$ is the discourse connectivity of the context with MWE component words (as shown with the complete graph in Figure 1); $c(G')$ is the discourse connectivity of the context without MWE component words (as shown with the subgraph $\{v_1, v_2, v_3\}$ in Figure 1).

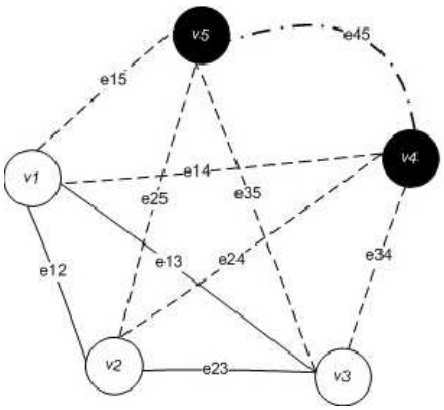

Figure 1: Cohesion Graph for identifying literal or non-literal usage of MWEs

4 Modeling Semantic Relatedness

In Section 3.1, we did not define how we model the semantic relatedness between two tokens ($h(t_i, t_j)$). Modeling semantic relatedness between two terms is currently an area of active research. There are two main approaches. Methods based on manually built lexical knowledge bases, such as WordNet, compute the shortest path between two concepts in the knowledge base and/or look at word overlap in the glosses (see Budanitsky and Hirst (2006) for an overview). Distributional approaches, on the other hand, rely on text corpora, and model relatedness by comparing the contexts in which two words occur, assuming that related words occur in similar context (e.g., Hindle (1990), Lin (1998a), Mohammad and Hirst (2006)). More recently, there has also been research on using Wikipedia and related resources for modeling semantic relatedness (Ponzetto and Strube, 2007; Zesch et al., 2008).

WordNet-based approaches are unsuitable for our purposes as they only model so-called "classical relations" like hypernymy, antonymy etc. For our task, we need to model a wide range of relations, e.g., between *ice* and *water*. Hence we opted for a distributional approach. We experimented with two different approaches, one (DV) based on syntactic co-occurrences in a large text corpus and the other (NGD) based on search engine page counts.

Dependency Vectors (DV) is a distributional approach which does not look simply at word co-occurrences in a fixed-size window but takes into account syntactic (dependency) relations between words (Padó and Lapata, 2007). Each target word is represented by a co-occurrence vector where dimension represents a chosen term and the vector contains the co-occurrence information between that word and the chosen terms in a corpus (we used the BNC in our experiments). A variety of distance measures can be used to compute the similarity of two vectors; here we use the cosine similarity which is defined as:

$$sim_{cos}(\overrightarrow{x}, \overrightarrow{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \quad (14)$$

Normalized Google Distance (NGD) uses the page counts returned by a search engine as proxies for word co-occurrence and thereby quantifies the strength of a relationship between two words (see Cilibrasi and Vitanyi (2007)). The basic idea is that the more often two terms occur together relative to their overall occurrence the more closely they are related. NGD is defined as follows:

$$NGD(x, y) = \frac{max\{log\ f(x), log\ f(y)\} - log\ f(x, y)}{log\ M - min\{log\ f(x), log\ f(y)\}} \quad (15)$$

where x and y are the two words whose association strength is computed, $f(x)$ is the page count returned by the search engine for the term x (and likewise for $f(y)$ and y), $f(x, y)$ is the page count returned when querying for "x AND y" (i.e., the number of pages that contain both, x and y), and M is the number of web pages indexed by the search engine. When querying for a term we query for a disjunction of all its inflected forms.[4] As it is difficult to obtain a specific and reliable number for the number of pages indexed by a search engine, we approximated it by setting it to the number of hits obtained for the word *the*. The assumption is that the word *the* occurs in all English language web pages (Lapata and Keller, 2005).

Using web counts rather than bi-gram counts from a corpus as the basis for computing semantic relatedness has the advantage that the web is a significantly larger database than any compiled corpus, which makes it much more likely that we can find information about the concepts we are looking for (thus alleviating data sparseness). However, search engine counts are notoriously unreliable (Kilgariff, 2007; Matsuo et al., 2007) and while previous studies have shown that web counts can be used as reliable proxies for corpus-based counts for some applications (Zhu and Rosenfeld, 2001; Lapata and Keller, 2005) it is not clear that this also applies when modeling semantic relatedness. We thus carried out a number of experiments testing the reliability of page counts (Section 4.1) and comparing the NGD measure to a standard distributional approach (Section 4.2).

[4] The inflected forms were generated by applying the *morph* tools developed at the University of Sussex (Minnen et al., 2001) which are available at: `http://www.informatics.susx.ac.uk/research/groups/nlp/carroll/morph.html`

4.1 Search Engine Stability

We first carried out some experiments to test the stability of the page counts returned by two of the most widely-used search engines, Google and Yahoo. For both search engines, we found a number of problems.[5]

Total number of pages indexed The total number of the web pages indexed by a search engine varies across time and the numbers provided are somewhat unreliable. This is a potential problem for NGD because we need to fix the value of M in Formula 15. As an approximative solution, we set it to the number of hits obtained for the word *the*, assuming that it will occur in all English language pages (Lapata and Keller, 2005).

Page count variation The number of page hits for a given term also varies across time (see example (4.1) for two queries for *Jim* at different times $t1$ and $t2$). However, we found that the variance in the number of pages tends to be relatively stable over *short* time spans, hence we can address this problem by carrying out all queries in one quick session without much delay. However, this means we cannot store page counts in a database and reuse them at a later stage; for each new example which we want to classify at a later stage, we have to re-compute all relevant counts.

(16) Hits(Jim, t1) = 763,000,000
Hits(Jim, t2) = 757,000,000

Problems with conjunction and disjunction The search engines' AND and OR operators are problematic and can return counter-intuitive results (see Table 1). This is a potential problem for us because we have to query for conjunctions of terms and disjunctions of inflected forms. For the time being we ignored this problem as it is not straightforward to solve.

	OPT = AND	*OPT* = OR
car	3,590,000,000	
car *OPT* car	4,670,000,000	3,550,000,000
car *OPT* car *OPT* car	3,490,000,000	3,530,000,000

Table 1: Operator test for Yahoo

Problems with high-frequency terms We also found that both the Google and Yahoo API seem to have problems with high frequency words, with the Google SOAP API throwing an exception and the Yahoo API returning the same 10-digit number for every high frequency word. This might be a data overflow problem. We addressed this problem by excluding high frequency words.

When comparing Yahoo and Google we found that Yahoo's page counts tend to be more consistent than Google's. We therefore opted for Yahoo in our further experiments.

4.2 NGD vs. Co-occurrence Vectors

In principle, we believe that the web-based approach for computing relatedness is more suitable for our task since it gives us access to more data and allows us to also model relations based on (up-to-date) world knowledge. However, the question arises whether the stability problems observed in the previous section have a negative effect on the performance of the NGD measure. To test this, we conducted a small study in which we compared the relatedness scores obtained by NGD and the semantic vector space model to the human ratings compiled by Finkelstein et al. (2002).[6]

We used Spearman's correlation test (Spearman, 1904) to compare the ranked human ratings to the ranked ratings obtained by NGD and the vector space method. The (human) inter-annotator agreement varies a lot for different pairs of annotators (between 0.41 and 0.82 by Spearman's correlation test), suggesting that deciding on the semantic relatedness between arbitrary pairs of words is not an easy task even for humans. In general, the NGD-human agreement is comparable to the human-human agreement. The agreement between the NGD and average human agreement is higher than some human-human agreements. Furthermore, we found that NGD actually outperforms the dependency vector method on this data set.[7] Hence, we decided to use NGD in the following experiments.

5 Experiments

We tested our graph-based classifiers on a manually annotated data set, which we describe in Sec-

[5] See also the discussions in Jean Véronis blog: `http://aixtal.blogspot.com` and the comments in Kilgariff (2007).

[6] The data sets are available at: `http://www.cs.technion.ac.il/~gabr/resources/data/wordsim353/`

[7] There may be several reasons for this. Apart from the fact that NGD has access to a larger data set, it may also be that syntactic co-occurrence information is not ideal for modeling this type of relatedness; co-occurrence information in a fixed window might be more useful. Furthermore, we did not spend much time on finding an optimal parameter setting for the dependency vector method.

tion 5.1. We report on our experiments and results in Section 5.2.

5.1 Data

Throughout the experiments we used the data set from Sporleder and Li (2009). The data consist of 17 potentially idiomatic expressions from the English Gigaword corpus, which were extracted with five paragraphs of context and manually annotated as 'literal' or 'non-literal' (see Table 2). The inter-annotator agreement on a doubly annotated sample of the data was 97% and the kappa score 0.7 (Cohen, 1960).

expression	literal	non-lit.	all
back the wrong horse	0	25	25
bite off more than one can chew	2	142	144
bite one's tongue	16	150	166
blow one's own trumpet	0	9	9
bounce off the wall*	39	7	46
break the ice	20	521	541
drop the ball*	688	215	903
get one's feet wet	17	140	157
pass the buck	7	255	262
play with fire	34	532	566
pull the trigger*	11	4	15
rock the boat	8	470	478
set in stone	9	272	281
spill the beans	3	172	175
sweep under the carpet	0	9	9
swim against the tide	1	125	126
tear one's hair out	7	54	61
all	862	3102	3964

Table 2: Idiom statistics (* indicates expressions for which the literal usage is more common than the non-literal one)

5.2 The Influence of Context Size and Weighting Scheme

To gain some insights into the performance of the graph-based classifier, we experimented with different context sizes and weighting schemes. In addition to the basic cohesion graph approach with five paragraphs of context (CGA), we tested a variant which only uses the current paragraph as context (CGA$_{para}$) to determine how sensitive the classifier is to the context size. We also experimented with three weighting schemes. The basic classifier (CGA) uses uniform edge and node weights. CGA$_{ew}$ uses edge weights based on the inverse distance between the tokens. CGA$_{nw}$ uses node weights based on idf. Finally, CGA$_{ew+nw}$ uses both edge and node weights.

We also carried out a pruning experiment in which we removed nodes from the graph that are only weakly connected to the context (called *weak cohesion nodes*). We hypothesize that these do not contribute much to the overall connectivity but may add noise. Pruning can thus be seen as a more gentle version of node weighting, in which we only remove the top n outliers rather than reweight all nodes. For comparison we also implemented a baseline (BASE), which always assigns the majority class ('non-literal').

Table 3 shows the results for the classifiers discussed above. In addition to accuracy, which is not very informative as the class distribution in our data set is quite skewed, we show the precision, recall, and F-score for the minority class (literal). All classifiers obtain a relatively high accuracy but vary in the precision, recall and F-Score values.

Method	LPrec.	LRec.	$LF_{\beta=1}$	$Acc.$
Base	–	–	–	0.78
CGA	0.50	0.69	0.58	0.79
CGA$_{para}$	0.42	0.67	0.51	0.71
CGA$_{prun}$	0.49	0.72	0.58	0.78
CGA$_{ew}$	0.51	0.63	0.57	0.79
CGA$_{nw}$	0.48	0.68	0.56	0.77
CGA$_{ew+nw}$	0.49	0.61	0.54	0.78

Table 3: Accuracy (Acc.), literal precision (LPrec.), recall (LRec.), and F-Score (LF$_{\beta=1}$) for the classifier

It can be seen that the basic cohesion graph classifier (CGA) outperforms the baseline on accuracy. Moreover, it is reasonably good at identifying literal usages among the majority of non-literal occurrences, as witnessed by an F-score of 58%. To obtain a better idea of the behavior of this classifier, we plotted the distribution of the MWE instances in the classifier's feature space, where the first dimension represents the discourse connectivity of the context with MWE component words ($c(G)$) and the second represents the discourse connectivity of the context without MWE component words ($c(G^{'})$). The graph-based classifier, which calculates the connectivity gain (see Equation 13), is a simple linear classifier in which the line $y = x$ is chosen as the decision boundary. Examples above that line are classified as 'literal', examples below as 'non-literal'. Figure 2 shows the true distribution of literal and non-literal examples in our data set. It can be seen that most non-literal examples are indeed below the line while most literal ones are above it (though a certain number of literal examples can also be found be-

low the line). So, in general we would expect our classifier to have a reasonable performance.

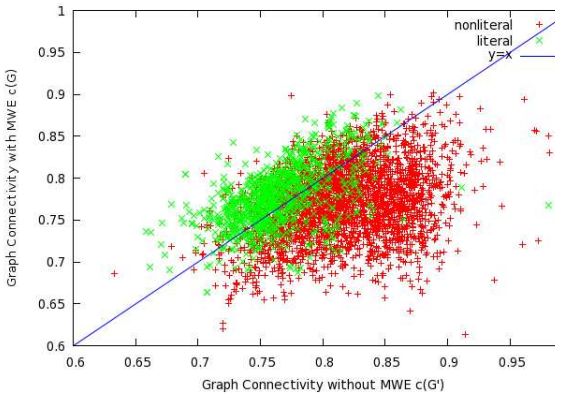

Figure 2: Decision boundaries of the cohesion graph

Returning to the results in Table 3, we find that a smaller context worsens the performance of the classifier (CGA$_{para}$). Pruning the 3 least connected nodes (CGA$_{prun}$) does not lead to a significant change in performance. Edge weighting (CGA$_{ew}$), node weighting (CGA$_{nw}$) and their combination (CGA$_{ew+nw}$), on the other hand, seem to have a somewhat negative influence on the literal recall and F-score. It seems that the weighting scheme scales down the influence of MWE component words. As a result, the product of the weight and the relatedness value for the idiom component words are lower than the average, which leads to the negative contribution of the idiom words to the cohesion graph (over predicting non-literal usage). We need to investigate more sophisticated weighting schemes to assign better weights to idiom component words in the future. The negative performance of the weighting scheme may be also due to the fact that we used a relatively small context of five paragraphs.[8] Both the idf and the distance weighting should probably be defined on larger contexts. For example, the distance between two tokens within a paragraph probably has not such a large effect on whether their relatedness score is reliable or accidental. Hence it might be better to model the edge weight as the distance in terms of paragraphs rather than words. The idf scores, too, might be more reliable if more context was used.

[8]Note that we used news texts which typically have very short paragraphs.

6 Conclusion

In this paper, we described an approach for token-based idiom classification. Our approach is based on the observation that literally used expressions typically exhibit strong cohesive ties with the surrounding discourse, while idiomatic expressions do not. Hence, idiomatic use of MWEs can be detected by the absence of such ties.

We propose a graph-based method which exploits this behavior to classify MWEs as literal or non-literal. The method compares how the MWE component words contribute the overall semantic connectivity of the graph. We provided a formalization of the graph and experimented with varying the context size and weighting scheme for nodes and edges. We found that the method generally works better for larger contexts; the weighting schemes proved somewhat unsuccessful, at least for our current context size. In the future, we plan to experiment with larger context sizes and more sophisticated weighting schemes.

Acknowledgments

This work was funded by the Cluster of Excellence "Multimodal Computing and Interaction".

References

T. Baldwin, A. Villavicencio. 2002. Extracting the unextractable: a case study on verb-particles. In *Proc. of CoNLL-02*.

T. Baldwin, E. M. Bender, D. Flickinger, A. Kim, S. Oepen. 2004. Road-testing the english resource grammar over the british national corpus. In *Proc. LREC-04*, 2047–2050.

C. Bannard, T. Baldwin, A. Lascarides. 2003. A statistical approach to the semantics of verb-particles. In *Proc. ACL 2003 Workshop on Multiword Expressions*.

J. Birke, A. Sarkar. 2006. A clustering approach for the nearly unsupervised recognition of nonliteral language. In *Proceedings of EACL-06*.

F. Bond, S. Shirai. 1997. Practical and efficient organization of a large valency dictionary. In *Workshop on Multilingual Information Processing Natural Language Processing Pacific Rim Symposium*.

A. Budanitsky, G. Hirst. 2006. Evaluating WordNet-based measures of semantic distance. *Computational Linguistics*, 32(1):13–47.

R. L. Cilibrasi, P. M. Vitanyi. 2007. The Google similarity distance. *IEEE Trans. Knowledge and Data Engineering*, 19(3):370–383.

J. Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurements*, 20:37–46.

P. Cook, A. Fazly, S. Stevenson. 2007. Pulling their weight: Exploiting syntactic forms for the automatic identification of idiomatic expressions in context. In *Proceedings of the ACL-07 Workshop on A Broader Perspective on Multiword Expressions*.

S. Evert, B. Krenn. 2001. Methods for the qualitative evaluation of lexical association measures. In *Proc. ACL-01*.

A. Fazly, P. Cook, S. Stevenson. 2009. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, 35(1):61–103.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, E. Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

L. Gerber, J. Yang. 1997. Systran mt dictionary development. In *Proc. Fifth Machine Translation Summit*.

M. Halliday, R. Hasan. 1976. *Cohesion in English*. Longman House, New York.

D. Hindle. 1990. Noun classification from predicate-argument structures. In *Proceedings of ACL-90*, 268–275.

G. Hirst, D. St-Onge. 1998. Lexical chains as representations of context for the detection and correction of malapropisms. In C. Fellbaum, ed., *WordNet: An electronic lexical database*, 305–332. The MIT Press.

R. Jackendoff. 1997. *The Architecture of the Language Faculty*. MIT Press.

G. Katz, E. Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *Proceedings of the ACL/COLING-06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*.

A. Kilgariff. 2007. Googleology is bad science. *Computational Linguistics*, 33(1):147–151.

M. Lapata, F. Keller. 2005. Web-based models for natural language processing. *ACM Transactions on Speech and Language Processing*, 2:1–31.

D. D. Lewis, W. B. Croft. 1990. Term clustering of syntactic phrase. In *Proceedings of SIGIR-90, 13th ACM International Conference on Research and Development in Information Retrieval*.

D. Lin. 1998a. Automatic retrieval and clustering of similar words. In *Proceedings of ACL-98*.

D. Lin. 1998b. Using collocation statistics in information extraction. In *Proc. MUC-7*.

D. Lin. 1999. Automatic identification of non-compositional phrases. In *Proceedings of ACL-99*, 317–324.

Y. Matsuo, H. Tomobe, T. Nishimura. 2007. Robust estimation of google counts for social network extraction. In *AAAI-07*.

G. Minnen, J. Carroll, D. Pearce. 2001. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223.

S. Mohammad, G. Hirst. 2006. Distributional measures of concept-distance: A task-oriented evaluation. In *Proceedings of EMNLP-06*.

S. Padó, M. Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

S. S. L. Piao, P. Rayson, D. Archer, A. Wilson, T. McEnery. 2003. Extracting multiword expressions with a semantic tagger. In *Proc. of the ACL 2003 Workshop on Multiword Expressions*, 49–56.

S. P. Ponzetto, M. Strube. 2007. Knowledge derived from Wikipedia for computing semantic relatedness. *Journal of Artificial Intelligence Research*, 30:181–212.

S. Riehemann. 2001. *A Constructional Approach to Idioms and Word Formation*. Ph.D. thesis, Stanford University.

T. T. Rila Mandala, H. Tanaka. 2000. Query expansion using heterogeneous thesauri. *Inf. Process. Manage.*, 36(3).

I. A. Sag, T. Baldwin, F. Bond, A. Copestake, D. Flickinger. 2001. Multiword expressions: a pain in the neck for NLP. In *Lecture Notes in Computer Science*.

G. Salton, J. Allan, C. Buckley, A. Singhal. 1994. Automatic analysis, theme generation and summarization of machine-readable texts. *Science*, 264(3):1421–1426.

F. Smadja. 19993. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–177.

C. Spearman. 1904. The proof and measurement of association between two things. *Amer. J. Psychol*, 72–101.

C. Sporleder, L. Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. In *Proceedings of EACL-09*.

N. Wacholder, P. Song. 2003. Toward a task-based gold standard for evaluation of NP chunks and technical terms. In *Proc HLT-NAACL*.

T. Zesch, C. Müller, I. Gurevych. 2008. Using wiktionary for computing semantic relatedness. In *Proceedings of AAAI-08*, 861–867.

X. Zhu, R. Rosenfeld. 2001. Improving trigram language modeling with the world wide web. In *Proceedings of ICASSP-01*.