An Unsupervised Approach to Interpreting Noun Compounds

Su Nam Kim
Dept. of Computer Science and Software Engineering
Carlton, VIC, AUstralia
snkim@csse.unimelb.edu.au

Timothy Baldwin
Dept. of Computer Science and Software Engineering
Carlton, VIC, AUstralia
tim@csse.unimelb.edu.au

Abstract

This paper proposes an unsupervised approach to automatically interpret noun compounds using semantic similarity. Our proposed unsupervised method is based on obtaining a large amount of robust evidence for NC interpretation. In order to obtain evidence sentences for semantic relations (SRs), we first acquired sentences containing both a head noun and its modifier in the form of SR definitions. Then we determined the semantic relations represented in the sentences by looking at the nouns in the test instances (noun mapping) and verbs in the SR definitions (verb mapping). In the noun mapping, we measured the similarity between nouns in test instances and nouns in the collected sentences. In the verb mapping, we mapped the verbs of sentences onto those in the SR definitions. Finally, we built a statistical classifier to interpret noun compounds and evaluated it over 17 SRs defined in [1].

Keywords:*Noun compound, Interpretation, Unsupervised approach*

1. Introduction

In the past few years noun compounds (or NCs) such as *computer science* and *paper submission* have received increasing attention as researchers work towards the goal of full text understanding. Semantic relations (or SRs), put simply, describe the nature of semantic association between the elements of the noun compound. For example, the semantic relation in *orange juice* expresses the fact that the modifier *orange* is a material or object used to make *juice* (the head noun). On the other hand, that of *morning juice* indicates that the head noun *juice* has specific significance to a point or internal in time (i.e. *morning*).

The primary computational challenges relating to NCs are:

- disambiguating syntax, i.e. bracketing [16, 14, 18]
- developing a set of SRs [15, 3, 26, 1, 24, 17, 22]
- interpreting SRs in NCs [26, 24, 17, 7, 8, 19, 4, 11]
- disambiguating word senses of components in NCs[9].

We will review previous work on NCs in Section 2.

In this paper, we are interested in developing an unsupervised method to automatically interpret NCs over a rich set of SRs. There has been little research on NC interpretation using unsupervised methods, with [12] and [13] being isolated examples of using corpus/web counts over predefined templates associated with different SR types to disambiguate NCs. Our attempt here is somewhat similar to that of [13] in terms of the attempt to robustly acquire large amounts of evidence for the interpretation task. However, in our case, we attempt to interpret NCs over a larger, more fine-grained set of SRs. Moreover, our focus is on utilizing the outcomes of previous studies (analogy-base [17, 7, 4] and underlying predicate [8, 19]) in order to collect reliable evidence for the unsupervised method.

We obtained the evidence (i.e. *evidence sentences*) through two different methods: verb mapping and noun mapping. Evidence sentences are extracted in relational form from the SR definitions, and include the head noun, modifier, sentential components and the voice of the verb. Verb mappings are based on the underlying predicate method of [8] while noun mappings follows the same basic intuition as the analogy-based method of [17, 10].

Our method works as follows. We first take a parsed corpus and extract sentences which fully or partially match the sentential definitions of each SR. Then we determine which SR is represented in the collected sentences using similarity measures. Evidence sentences can contain either both a head noun and modifier but a verb different from that in an SR definition (i.e. something other than a *definition verb*), or a definition verb but one or both nouns not matching those in the test NCs (i.e. the set of NCs that we wish to automatically interpret). In the first case, we map corpus-attested verbs onto definition verbs via a verb mapping method, while in the latter case, we map corpus-attested noun pairs onto test NCs via a noun mapping method. Once we have collected and interpreted the evidence sentences, we build a classifier to automatically interpret NCs based on their combined evidence.

In the following sections, we present an overview of previous work relating to NCs (Section 2), detail our motivation (Section 3), provide a detailed description of the underlying approach (Section 4), detail the data and resources (Section 5), present the experimental results and analysis (Section 6), and finally describe our conclusions (Section 8).

2. Related Work

Interpreting NCs via semantic relations has been studied from both linguistic and computational perspectives. Such studies have led to the identification of differing sets of possible SRs, often formulated from different research perspectives [15, 3, 26, 23]. The majority of studies (e.g. [15, 25]) have proposed a small number of generalisable SRs, while [3] suggested that the set of SRs is unbounded. The main issues which shape a given set of SRs are granularity, coverage and distribution of instances among SRs, as discussed by [23]. Since the set of SRs directly influences the ability of classifiers to automatically interpret NCs, it is important to arrive at a standardized set of SRs. Unfortunately this is still an issue of considerable debate.

Another area of concerted research has been the development of automated methods for discerning the syntactic and semantic structure of NCs. The predominance of research has been on supervised methods, with varying degrees of success [17, 5, 6, 7, 8, 20, 19, 4] The two main supervised approaches are the semantic similarity (or analogy-based) method [24, 17, 5, 7, 20, 4, 10, 23] and the underlying predicate method (i.e. semantic disambiguation relative to an underlying predicate or semantically-unambiguous paraphrase) [12, 21, 8, 19]. The first approach uses the semantics of the component nouns, while the latter uses semantics of ellipsed predicates between the head noun and its modifier(s) [11].

3. Motivation and Method

The basic intuition behind this work is that we can interpret NCs by looking at their corpus co-occurrence in sentential contexts since the interpretations of NCs are often present in the corpus. For example, the sentence, *the printer is located in the lab* indicates that SR contained in the NC, *lab printer* can be interpreted as LOCATION, i.e. the *printer* is located in the *lab*. Hence, we can directly use these sentences as evidence to interpret NCs without additional training data.

Naturally, life is not this simple, in that we do not normally find enough evidence sentences to determine the SR for a given noun pairing with sufficient confidence. We often find evidence sentences with missing information: sentences in relational form that contain both the head noun and modifier but without a verb which matches any of the verbs in the SR definitions; and sentences with verbs from the SR definitions where one or both of the nouns is not a test NC example. For example, *the bulb generates light* supports an interpretation of the NC *light bulb* as corresponding to the SR PRODUCT, (i.e. the *bulb* produces *light*). However, this sentence does not provide direct evidence for PRODUCT as the verb *generate* does not match any of the definition verbs for PRODUCT. On the other hand, when we attempt to interpret the NC *Toyota Camry*, the sentence *The Camry is manufactured by Toyota* is also excluded as evidence for PRODUCT since the head noun *Camry* in *Toyota Camry* does not match with *car* in the test NC *Toyota car*. However, according to our previous work [10], similar nouns tend to have the same SR (e.g. the SR of *apple juice* and *orange nectar* is the same due to component similarity.) We can exploit this to interpret *Toyota car* by checking its similarity with *Toyota Camry*.

In order to interpret NCs without supervision, we focus on acquiring evidence sentences which would otherwise be ignored due to mismatches with the definition sentences. Our main effort is on extracting information from these sentences by mapping either the verbs or nouns to those in the SR definitions using lexical similarity. The details are described in Sections 3.1 and 3.2.

3.1. Verb Mapping Method

Figure 1 is an illustration of the verb mapping method, where we compare a verb not contained in any of the SR definitions against the set of definition verbs for each SR. First, we collect sentences which match the definition sentences of SRs and have both a head noun and modifier in the test NCs. Next, for remaining instances where we have

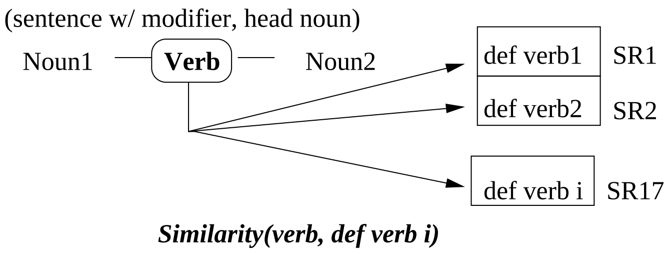

Figure 1. Verb mapping method

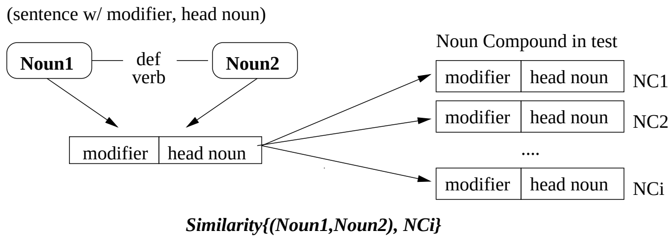

Figure 2. Noun mapping method

a test NC head noun and modifier but the verb isn't a definition verb, we compare the verb with each of the definition verbs. In Figure 1, *noun1* and *noun2* are the modifier and head noun respectively, and *verb* is a non-definition verb. We map *verb* onto the definition verb(s) using lexical similarity.

This method is identical to the verb mapping method (i.e. verb mapping by `WordNet::Similarity`) introduced by us in previous work [8]. We use this method to acquire evidence sentences for unsupervised NC interpretation. We also use `WordNet::Similarity` to measure the similarity between the actual verb and definition verbs, and replace the actual verb with closest matching (1-nearest neighbour) definition verb. Take, for example, the sentence, *the bulb generates light*. Since the verb, *generate* does not appear in the list of definition verbs, we search for the closest-matching definition verb by measuring the similarity between *generate* and each of the definition verbs. Finally, we select the most similar definition verb (*produce*) as a substitute for the original verb. The resulting sentence *the bulb produces light* can now be used as evidence for interpreting *light bulb* as PRODUCT.

Note that use two methods to arrive at a type-level interpretation for each NC based on its token occurrences: (1) *simple count* of the sentences matched to each SR, and (2) the *weighted count* of token instances (as used in [8]) based on the prior of a given verb co-occurring, according to Equation (1):

$$Weight(V_j) = \frac{\sum_{i=1}^{n}(H_i, V_j)}{\sum_{k=1}^{m}\sum_{i=1}^{n}(H_i, V_k)} \qquad (1)$$

where V_j is a given definition verb and H_i is each head noun.

3.2. Noun Mapping Method

Figure 2 depicts the noun mapping method, which operates over sentences with noun(s) other than those we are attempting to classify directly (i.e. other than the test NCs). The motivation behind the noun mapping method is that the sense pairing of the head noun and modifier determines the SR of the NC. The method comes from [17] originally, but was extended by us in earlier work [10] to include similar nouns, in the form of synonyms, hypernyms and sister nouns. Hence, when we have sentences that do not include the test NCs but noun pairings similar to them, we can still use them as extra evidence to interpret the test NCs.

In our approach, we first collect sentences containing definition verbs with (common) nouns in each of two pre-defined grammatical roles (e.g. subject, object or object of a selected-for preposition). If one or both of the nouns do not match those in the test NCs, we compare each noun to its corresponding noun (based on its grammatical role) in a test NC using `WordNet::Similarity`. Once the closest-matching NC has been found, we replace the nouns in the original sentence with those in the test NC. As a result of this step, we derive a set $\{S_{SR_i,j}\}$ of collected sentences for each test NC for each SR from a given original sentence j. For example, *this pie is made of apples* can be mapped onto the NC *apple pie*. *Apple pie* is not a test NC, but the test NC *pecan pie* can be identified as a close match. After applying the noun mapping, we get the modified sentence *this pie is made of pecans*. Note that as part of the noun mapping, we apply a threshold to filter out sentences where the similarity with a test NC is not sufficiently high.

To derive a type-level prediction for an NC based on token-level evidence sentences, we use three methods: (1) *simple count*, (2) *normalized count*, and (3) *normalized similarity*. The *normalized count* for a given SR ($N(SR_i)$ in Equation (2)) averages the simple count relative to the number of distinct relational forms associated with that SR, as the number of relational forms is not constant across SRs. For example, if the number of evidence sentences for *pecan pie* as MATERIAL were 5 and the number of relational forms for the SR MATERIAL were 3, then the final value would be $\frac{5}{3}$.

$$N(SR_i) = \frac{\sum_j |S_{SR_i,j}|}{|RF_{SR_i}|} \qquad (2)$$

The *normalized similarity* ($NSim$ in 3) takes the form of the weighted sum over the evidence sentences, based on the similarity between the original noun pairing and the test

NC:

$$NSim(SR_i) = \frac{\sum_j (|S_{SR_i,j}| \cdot sim_j)}{|RF_{SR_i}|} \quad (3)$$

Here, and in Equation (2), SR_i is a given SR, $S_{SR_i,j}$ is an original sentence j which has been mapped onto SR$_i$, and RF_{SR_i} is a relational form of SR_i.

4. System Architecture

Figure 3 depicts the architecture of our system. The first step is to collect relevant sentences from our corpus, for which purpose we first parse the corpus with RASP [2] to generate the predicate–argument structure for each verb in terms of its (deep) subject, (deep) object(s), and the nominal object of any selected-for prepositions; we additionally analyse the voice of each sentence. As our corpus we use the British National Corpus, Brown Corpus and Wall Street Journal. Note that only verbs with two nominal arguments in total in any of the subject, object or prepositional object positions is used as a potential evidence sentence.

Potential evidence sentences which contain a definition verb and one of the test NCs in the appropriate configuration are first identified and put aside as evidence sentences. The remaining potential evidence sentences are then fed into the two mapping methods in order to acquire evidence for SR interpretation. Recall that the verb mapping deals with sentences where both the head noun and modifier coincide with a test NC but the verb is not a definition verb. Hence, we map the actual verbs onto the definition verbs by lexical similarity to increase the amount of evidence we can use in SR interpretation. For the noun mapping method, on the other hand, the sentences must include one of the definition verbs but not one of the test NCs (or at least not in the correct configuration). Here, we compare the head noun and modifier in the original sentence with each of the test NCs, and use the sentence as extra evidence for the closest-matching test NC being an instance of the SR defined by the verb. With both the verb mapping and the noun mapping, we use the wup, lin and vector metrics in `WordNet::Similarity`. Finally, we combine all of the evidence sentences from the various sources and predict the single most-likely SR for each test NC based on the combined body of evidence for each SR by summing up the (weighted or otherwise) counts.

5. Data and Resources

5.1. `WordNet::Similarity`

`WordNet::Similarity`[1] is an open-source package for calculating the lexical similarity between word (or sense) pairs based on a variety of similarity measures. The similarity measures are categorized into three groups: path-based (wup, lch), content-information based (jcn, lin) and relatedness (lesk, vector). As we were interested in using the similarity as a threshold to filter out evidence sentences with low similarity, we used wup, lin and vector, all of which are similarity measures which return a real-number value in the range $[0, 1]$. wup uses the path length from the least common subsumer of two concepts to the root note (Equation (4)). lin is based on the ratio of the information content of the least common subsumer, to the information content of each of the target concepts (Equation (5)). vector is a cosine-based similarity measure based on multi-dimensional vector analysis of the glosses associated with each word sense (Equation (6)):

$$Similarity_{wup}(c_1, c_2) = \frac{1}{p} \quad (4)$$

$$Similarity_{lin}(c_1, c_2) = \frac{2 \times IC(lcs(c_1, c_2))}{IC(c_1) + IC(c_2)} \quad (5)$$

$$Relation_{vector}(c_1, c_2) = \frac{\vec{v_1} \cdot \vec{v_2}}{|\vec{v_1}||\vec{v_2}|} \quad (6)$$

where (c_1, c_2) is a pairing of WordNet concepts, p is the number of nodes in the shortest path between c_1 and c_2, and $\vec{v_1}$ and $\vec{v_2}$ are the term vectors associated with the target concepts.

5.2. Semantic Relations

We used the set of SRs defined in [1], for consistency with previous work on the verb mapping method [8]. The full set of 20 SRs and definition sentences is presented in Table 1. Among the 20 SRs, we only used 17 SRs, as TIME, EQUATIVE and PROPERTY do not have any associated relational form. Note that [8] reported that TIME was easily detected by looking at the semantics of the modifier, whereas EQUATIVE was problematic to deal with since its definition takes the form of a conjunction (without a verb); it is also sufficiently low-frequency that we are not unduly biasing the performance of NC interpretation in excluding it.

5.3. Data Collection

Our set of test NCs and definition verbs (originally called *seed verbs*) were taken from previous work [8]. The number of test NCs is 88. Unlike our previous work, we used only 54 seed verbs (i.e. the small set from the original paper) as our definition verbs, in order to restrict the semantics of the SRs.

[1] http://www.d.umn.edu/ tpederse/similarity.html

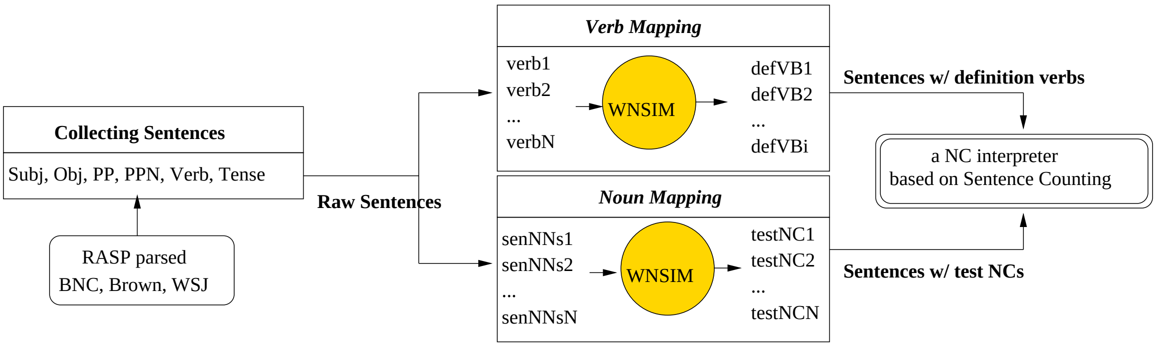

Figure 3. System architecture

	Measure		
M	WUP	LIN	VECTOR
Count	.217	.179	**.266**
Weight	.253	.155	.215

Table 2. Accuracy with the verb mapping **method**

As the source of our evidence sentences, we used the combination of the British National Corpus, Brown Corpus and Wall Street Journal.

The final number of sentences derived by the verb mapping method was 4,600, 4,051 and 1,666, using wup, lin and vector respectively. The evidence sentences were spread across all the test NCs, with at least five sentences per test NC.

We collected 9,178 sentences based on the noun mapping method. The final number of evidence sentences used in our method varied depending on our choice of threshold value (few sentences at a threshold value of 0.9; considerably more at a threshold value of 0.6).

6. Evaluation

In our evaluation, we calculated a baseline performance by estimated the performance of a random baseline via $\frac{1}{N}$, where N is the number of SRs we use in our evaluation (17, as we exclude TIME, PROPERTY and EQUATIVE). This leads to a baseline accuracy of .059.

Table 2 shows the performance of the unsupervised NC interpretation using the verb mapping method. *Count* is based on the simple count of the evidence sentences, while *Weight* based on the weighted sentence count, according to Equation (1).

Table 3 shows the performance of the noun mapping method over various threshold values, and the three counting methods: simple count, normalized count (see Equation (2)), and normalized similarity count (see Equation (3)).

7. Discussion

We first compared two mapping methods: verb mapping and noun mapping. All instances of the verb mapping method outperformed those of the noun mapping method. Our unsupervised NC interpretation method achieved the best performance with the vector measure using verb mapping (**.266** vs. **.153**). We observed that the verb mapping method produced a much smaller number of evidence sentences, but that they contained less errors (i.e. the error rate in the evidence collected by the verb mapping method is much lower). Hence, we conclude that the quality of collected evidence is a more important consideration than the raw amount of evidence.

Secondly, we compared the individual similarity measures. We observed a different pattern of performance across the two mapping methods. wup worked well with both mapping methods, whereas vector performed well with the verb mapping method but less so with the noun mapping method. Analysis of the collected evidence revealed that vector produced less evidence sentences than the other two measures, and when used with the noun mapping method, those sentences had high error rates. Hence, vector paired with the noun mapping method performed the worst due to it producing small amounts of high noise data. The computational complexity of vector is much higher than the other two methods, suggesting that wup is the best choice of similarity measure.

Thirdly, we compared the different methods for interpreting the counts associated with a given method. With the verb mapping method, we didn't observe a significant difference between *simple counts* and *weighted counts*: our system achieved an accuracy of .266 with simple counting and vector, and .253 with weighted counting and wup. On the other hand, we found that the noun mapping method

Relation	*Definition*	*Example*
AGENT	N_2 is performed by N_1	*student protest, band concert, military assault*
BENEFICIARY	N_1 benefits from N_2	*student price, charitable compound*
CAUSE	N_1 causes N_2	*printer tray, flood water, film music, story idea*
CONTAINER	N_1 contains N_2	*exam anxiety, overdue fine*
CONTENT	N_1 is contained in N_2	*paper tray, eviction notice, oil pan*
DESTINATION	N_1 is destination of N_2	*game bus, exit route, entrance stairs*
EQUATIVE	N_1 is also head	*composer arranger, player coach*
INSTRUMENT	N_1 is used in N_2	*electron microscope, diesel engine, laser printer*
LOCATED	N_1 is located at N_2	*building site, home town, solar system*
LOCATION	N_1 is the location of N_2	*lab printer, desert storm, internal combustion*
MATERIAL	N_2 is made of N_1	*carbon deposit, gingerbread man, water vapour*
OBJECT	N_1 is acted on by N_2	*engine repair, horse doctor*
POSSESSOR	N_1 has N_2	*student loan, company car, national debt*
PRODUCT	N_1 is a product of N_2	*automobile factory, light bulb, color printer*
PROPERTY	N_2 is N_1	*elephant seal, blue car, big house, fast computer*
PURPOSE	N_2 is meant for N_1	*concert hall, soup pot, grinding abrasive*
RESULT	N_1 is a result of N_2	*storm cloud, cold virus, death penalty*
SOURCE	N_1 is the source of N_2	*chest pain, north wind, foreign capital*
TIME	N_1 is the time of N_2	*winter semester, morning class, late supper*
TOPIC	N_2 is concerned with N_1	*computer expert, safety standard, horror novel*

Table 1. Semantic relations

	Simple Count			*Normalized Count*			*Normalized Similarity*		
Threshold	WUP	LIN	VECTOR	WUP	LIN	VECTOR	WUP	LIN	VECTOR
0.6	.094	.078	.059	.094	.091	.059	.094	.091	.059
0.7	**.130**	.085	.071	**.153**	.113	.071	**.153**	.099	.082
0.8	.094	.109	.025	.145	.109	.035	.133	.109	.049
0.9	.129	.121	.102	.148	.121	.143	.148	.121	.142

Table 3. Accuracy with the noun mapping **method**

performed better with *normalized counting* and *normalized similarity*. In terms of different thresholds, surprisingly, we did not attain conclusive results to confirm which threshold performs best. For vector and lin, there is a roughly constant dropoff in accuracy as we lower the threshold (and hence introduce greater amounts of noise), but with wup, a threshold value of 0.7 was the sweet spot in terms of quality and quantity of data across all three counting methods.

Finally, we compared our method with other existing methods. Unfortunately, we do not have an unsupervised benchmark system which runs over the full set of SRs to directly compare our method to. However, since we used the same test data and the same set of SRs as [8], we can compare the performance with this earlier supervised method. [8] achieved an accuracy of .526 with *weighted counting* and vector, and a baseline accuracy based on majority vote (i.e. zero-R) of .423. Unsurprisingly, compared to our unsupervised method, the supervised methods performed much better over the same data set. From this, we can only conclude that it is much harder to achieve good performance with an unsupervised approach to NC interpretation even when the underlying method (i.e. verb mapping) is the same.

8. Conclusion

We proposed an unsupervised method for automatically interpreting NCs. We introduced two mapping methods to collect evidence: verb mapping and noun mapping. We tested the verb mapping and noun mapping methods over three similarity measures using `WordNet::Similarity`. Finally, we evaluated our method over 17 SRs defined in [1], and found the verb mapping method performed the best through producing lower quantities of higher quality evidence sentences.

As our contribution in this work, we proposed a novel unsupervised method to automatically interpret NCs over a full scale set of SRs. Despite the poor performance of our method compared to previously-published supervised methods, we believe that NC interpretation using unsupervised approaches has room for improvement and definitely warrants further attention. We also made use of previous research in formulating our evidence collection methods.

References

[1] K. Barker and S. Szpakowicz. Semi-automatic recognition of noun modifier relationships. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING-1998)*, pages 96–102, Montreal, Canada, 1998.

[2] T. Briscoe and J. Carroll. Accurate statistical annotation of general text. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*, pages 1499–1504, Las Palmas, Canary Islands, 2002.

[3] T. W. Finin. *The semantic interpretation of compound nominals*. PhD thesis, University of Illinois, Urbana Champaign, 1980.

[4] R. Girju. Improving the interpretation of noun phrases with cross-linguistic information. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 568–575, Prague, Czech Republic, 2007.

[5] R. Girju, D. Moldovan, M. Tatu, and D. Antohe. On the semantics of noun compounds. *Computer Speech and Language*, 19(44):479–496, 2005.

[6] C. Grover, M. Lapata, and A. Lascarides. A comparison of parsing technologies for the biomedical domain. *Journal of Natural Language Engineering*, 1(1):1–38, 2004.

[7] S. N. Kim and T. Baldwin. Automatic interpretation of compound nouns using WordNet similarity. In *Proceedings of 2nd International Joint Conference on Natual Language Processing (IJCNLP-2005)*, pages 945–956, Jeju, Korea, 2005.

[8] S. N. Kim and T. Baldwin. Interpreting semantic relations in noun compounds via verb semantics. In *Proceedings of Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics and 21st International Conference on Computational Linguistics (COLING/ACL-2006)*, pages 491–498, Sydney, Australia, 2006.

[9] S. N. Kim and T. Baldwin. Disambiguating noun compounds. In *Proceedings of 22nd AAAI Conference on Artificial Intelligenc*, pages 901–906, Vancouver, Canada, 2007.

[10] S. N. Kim and T. Baldwin. Interpreting noun compounds using bootstrapping and sense collocation. In *Proceedings of Conference of the Pacific Association for Computational Linguistics*, pages 129–136, Melbourne, Australia, 2007.

[11] S. N. Kim and T. Baldwin. Benchmarking noun compound interpretation. In *Proceedings of 3rd International Joint Conference on Natual Language Processing (IJCNLP-2008)*, pages 569–576, Hyderabad, India, 2008.

[12] M. Lapata. The disambiguation of nominalizations. *Computational Linguistics*, 28(3):357–388, 2002.

[13] M. Lapata and F. Keller. The web as a baseline: Evaluating the performance of unsupervised web-based models for a range of NLP tasks. In *Proceedings of the Human Langauge Techinology Conference and Conference on Empirical Methods in National Language Processing (HLT/NAACL-2004)*, pages 121–128, Boston, USA, 2004.

[14] M. Lauer. *Designing Statistical Language Learners: Experiments on Noun Compounds*. PhD thesis, Macquarie University, 1995.

[15] J. Levi. *The Syntax and Semantics of Complex Nominals*. Academic Press, New York, New York, USA, 1978.

[16] M. Marcus. *A Theory of Syntactic Recognition for Natural Language*. MIT Press, Cambridge, USA, 1980.

[17] D. Moldovan, A. Badulescu, M. Tatu, D. Antohe, and R. Girju. Models for the semantic classification of noun phrases. In *Proceedings of HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, pages 60–67, Boston, USA, 2004.

[18] P. Nakov and M. Hearst. Search engine statistics beyond the n-gram: Application to noun compound bracketting. In *Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 17–24, Ann Arbor, USA, 2005.

[19] P. Nakov and M. Hearst. Using verbs to characterize noun-noun relations. In *Proceedings of the 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA)*, pages 233–244, Bularia, 2006.

[20] V. Nastase, J. Sayyad-Shirabad, M. Sokolova, and S. Szpakowicz. Learning noun-modifier semantic relations with corpus-based and WordNet-based features. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI)*, pages 781–787, Boston, USA, 2006.

[21] J. Nicholson and T. Baldwin. Statistical interpretation of compound nominalisations. In *Proceedings of the Australian Language Technology Workshop*, pages 152–159, Sydney, Australia, 2005.

[22] D. Ó Séaghdha. Designing and evaluating a semantic annotation scheme for compound nouns. In *Proceedings of the 4th Corpus Linguistics Conference*, 2007.

[23] D. Ó Séaghdha and A. Copestake. Co-occurrence contexts for noun compound interpretation. In *Proceedings of the ACL-2007 Workshop on A Broader Perspective on Multiword Expressions*, pages 57–64, Prague, Czech Republic, 2007.

[24] B. Rosario and H. Marti. Classifying the semantic relations in noun compounds via a domain-specific lexical hierarchy. In *Proceedings of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP-2001)*, pages 82–90, Pittsburgh, Pennsylvania, USA, 2001.

[25] K. Sparck Jones. *Compound noun interpretation problems*. Prentice-Hall, Englewood Cliffes, USA, 1983.

[26] L. Vanderwende. Algorithm for automatic interpretation of noun sequences. In *Proceedings of the 15th Conference on Computational linguistics*, pages 782–788, Kyoto, Japan, 1994.