Interpreting Noun Compounds using Bootstrapping and Sense Collocation

Su Nam Kim and Timothy Baldwin
Department of Computer Science and Software Engineering
and
NICTA Victoria Lab
University of Melbourne, VIC 3010 Australia
{snkim,tim}@csse.unimelb.edu.au

Abstract

This paper describes a bootstrapping method for automatically tagging noun compounds with their corresponding semantic relations. Our work takes advantage of the collocation of senses of the noun compound constituents and also word similarity. We exploit this to generate a set of noun compounds from a set of previously tagged noun compounds by replacing one constituent of each noun compound with similar words that are derived from synonyms, hypernyms and sister words.

We started with 200 "seed" noun compounds and generated sets of derived noun compounds with accuracy ranging between 64.72% and 70.78%. We also evaluated the utility of the automatically derived noun compounds when used in combination with existing noun compound interpretation methods.

1 Introduction

Research on the semantics of noun compounds (NCs) has generally focused on either devising systems for interpreting NCs or alternatively automatically interpreting NCs according to a pre-defined set of semantic relations (SRs) (Levi, 1979; Lauer, 1995; Barker and Szpakowicz, 1998). In almost all previous work, the interpretation of semantic relations has been suggested as being highly relevant to NLP applications including question-answering and machine translation.

Linguistically-motivated systems for interpreting NCs vary considerably in the number of semantic relations used (Levi, 1979; Finin, 1980). Computational research on the topic has tended to take a predictably data-driven approach to the question and has combined the development of semantic relation sets with interpretation methods (Vanderwende, 1994; Barker and Szpakowicz, 1998).

There are two primary motivations underlying this work: (1) the suggestion by Jones (1983) and others that disambiguating the word senses of the component words in NCs is an essential component of interpretation, providing the motivation for this research; and (2) the recent trend in NC research towards automated methods that maximize use of small amounts of annotated data, recognizing that annotation is a significant bottleneck in NC research (Barker and Szpakowicz, 1998; Moldovan et al., 2004).

In this paper, we aim to interpret the semantic relations in NCs with a minimum annotation overhead. In this, we differ somewhat from previous research attempts, which have generally presupposed large amounts of training data. As a result, previous methods are not readily portable to new domains or well suited to redeployment over new languages with analogous constructions. Additionally, despite the reliance on large amounts of training data, results have tended to be modest, generally with an error rate of over 40% for representative sets of semantic relations.[1]

The approach discussed in this paper takes NCs such as *apple pie* and *family car*, and attempts to derive an interpretation in the form of the semantic relation between the modifier(s) and the head noun. Examples of such semantic relations include MAKE[2] (e.g. *apple pie* signifies a *pie* made from *apple(s)*) and POSSESSOR (e.g. *family car* signifies a *car* possessed by a *family*).

In our method, we use semi-supervised learning to automatically build up a collection of NCs

[1]Moldovan et al. (2004) claimed about 43% accuracy and Kim and Baldwin (2006) reported an accuracy slightly above 50%.

[2]The semantic relations are taken from Barker and Szpakowicz (1998).

tagged with semantic relations, using sense collocations of component words in NCs in the training data. That is, instead of manually tagging all training instances, we tag only a selected handful of seed instances and bootstrap off these by replacing the component words in training NCs with their similar nouns. Similarity is evaluated via ontological semantics, using three basic relation types: synonyms, hypernyms and sister words. Novel NCs which have been generated from training instances via similarity are then tagged according to the semantic relation of the original training instance. In this way, we automatically derive the semantic relations of novel NCs. More importantly, we employ the expanded set of training instances in NC interpretation and find that performance is significantly enhanced.

2 Motivation

Despite the infamous high productivity of noun compounds (c.f. the oft-cited *apple juice seat*), we claim that natural occurrences of NCs are often constrained by the semantic collocation of their constituents. That is, combinations of words from certain semantic classes are more likely to form NCs than others due to their ease of interpretation. Based on this observation, Moldovan et al. (2004) used semantic collocations of constituents to interpret NCs and showed that for NCs with the same sense collocation, the same semantic relation holds with high reliability. Similarly, in Kim and Baldwin (2005), the semantic similarities between training and test instances play an important role in interpreting NCs.

Based on these facts, we hypothesize that not only do given sense collocations in NCs lead to a consistent semantic relation, but semantically similar word combinations share the same semantic relations. For example, the semantic relation of *horse doctor* is OBJECT, as the modifier, *horse*, is acted on by the head noun, *doctor*.[3] When the modifier, *horse*, is replaced by its hypernym, *animal*, we are still able to correctly predict the semantic relation of *animal doctor* as OBJECT. This process can also be seen to apply to sister words. For example, the NC *orange juice* has the semantic relation MAKE that the head noun, *juice* is made from the modifier, *orange*. By replacing *orange* with its sister word, *lemon*, the newly generated NC *lemon juice* also has the same semantic relation, MAKE.

The following examples illustrate how different semantic relations, namely synonym, hypernym and sister words can share the same semantic relation.

(1) automobile factory

- **Synonym:** car factory
- **Hypernym:** vehicle factory
- **Sister:** truck factory

(2) automobile factory

- **Synonym:** automobile mill
- **Hypernym:** automobile plant
- **Sister:** automobile mint

In (1) and (2), we replace one noun at a time since we want to restrict the degree of semantic variation. In (1) we replace the modifier with related words, and in (2) we replace the head noun with related words. By replacing a constituent with a synonym, we generate *car factory* and *automobile mill*, both of which have exactly the same sense collocation as the original NC (as the replaced constituents belong to the same semantic class as the original words). On the other hand, *vehicle factory*, *automobile plant*, *truck factory*, and *automobile mint*, whose constituents are replaced by a hypernym in the first two instances and sister word in the second two instances, have different sense collocations to that original NC and yet maintain the same semantic relation.

Notice in (2) that with *automobile mint*, we generate what is a pragmatically-marked NC with a semantically-plausible interpretation. That is, our world knowledge about how mints operate and what they produce makes the generated NC sound unnatural, but if we can get beyond this to interpret the NC, the semantic relation we come up with is plausible. To deal with cases like this, our proposed method checks all generated NCs against a pre-compiled list of attested NCs and filters out anything which has not been observed in corpus data.

A second more serious concern is that we will produce an NC which violates our basic assumption about the semantic relation being preserved

[3] The example is taken from Barker and Szpakowicz (1998). Note that defining semantic relations is controversial. In this paper, we do not intend to define or discuss semantic relations. Instead, we decide to use the set of semantic relations in Barker and Szpakowicz (1998) without modification due to reliability shown in previous work.

across controlled semantic variation of the constituent words. For example, an alternate sister of *factory* is *recycling plant*, leading to *car recycling plant*. Here, the most natural interpretation is OBJECT rather than MAKE. We detail a filtering method for dealing with such examples in the next section.

3 Approach

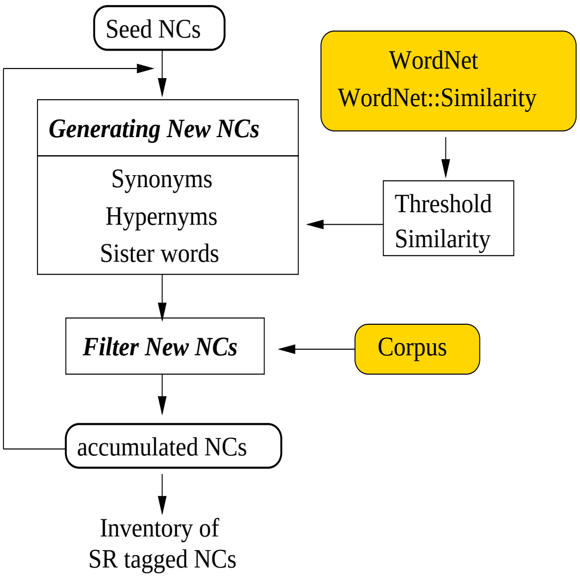

Figure 1: System Architecture

Figure 1 depicts our method for automatically deriving interpreted NCs. We initially begin with a small number of seed NCs that are manually tagged with semantic relations and word senses. We then generate new NCs by replacing each of the modifier and head noun with sense-specified words based on synonyms, hypernyms and sister words derived from WORDNET2.1. Note here that we only replace one noun at a time to avoid over-generation.

Next, we apply a filtering method based on word-level similarity. That is, if the similarity between the original noun and its substitution candidate is less than a predefined threshold, we discard the candidate. Note that the threshold is defined differently for the three ontological relation types. Figure 2 shows the distribution of similarity values across the three ontological relation types, as determined using the WUP (Wu and Palmer, 1994) similarity measure (see Section 4.2) across the cross product of 200 randomly extracted nouns. Table 1 shows the average similarity values and the threshold used in our method for each ontological relation type. The threshold is determined when the number of tested NCs is approximately 90%.

After gathering the generated NCs, we apply a corpus attestation-based filtering method to filter out false positives. Despite the high productivity of NCs, many of the automatically generated NCs are pragmatically or semantically implausible (e.g. *part part* or *activity being*). As we assume there is a default interpretation for a given NC which applies equally to all occurrences of that NC, this check is carried out independent of context.

Finally, we assign the semantic relation from the original NC to the newly-generated NCs derived from it. We then iteratively repeat this procedure over the newly-generated NCs until some termination condition is met, namely no new NCs are generated or a threshold on the number of iterations is met. The number of iterations for synonyms and sister words is limited to 1 since these two relations are commutative.[4] For hypernyms, the only limit on the number of iterations the algorithm could be run over is the depth of the WORDNET hierarchy, but in practice, we stop after the 4th iteration.

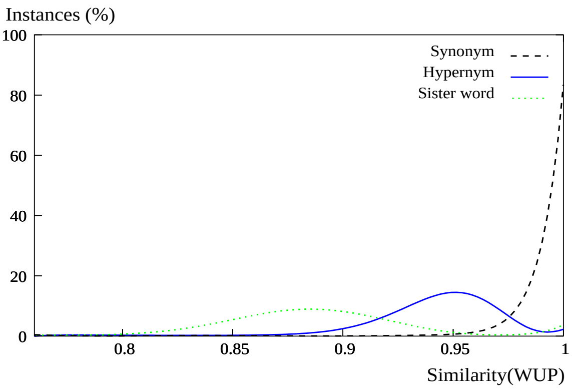

Figure 2: Range of similarity values over different ontological relation types

4 Data and Resources

4.1 Semantic Relations

Semantic relations specify the interpretation of the overall NC relative to it parts. For example, *winter school* is a *school* held during *winter* (semantic relation = TIME). The semantic relation is sensi-

[4]Strictly speaking, they are commutative at the word sense but not necessarily word level.

Relation	Average	Threshold
Synonym	.94	.90
Hypernym	.89	.85
Sister	.83	.80

Table 1: Average similarity and threshold for each ontological relation type

tive to the semantics of the NC parts. Although NCs have similar parts, they could have different semantic relations. For example, in *student discount*, *students* BENEFIT from the *discount*. On the other hand, in *student protest*, *students* are the AGENT of the action represented by *protest*.

As mentioned above, there is no consensus on the optimal number or set of semantic relations for English (or any other language). In this paper, rather than defining a new set of semantic relations, we simply (and naively) use the set of 20 semantic relations defined by Barker and Szpakowicz (1998) as each semantic relation is clearly defined and it has relatively high currency in computational linguistics research. Additionally, since Kim and Baldwin (2005) used the same set of semantic relations, we can directly compare our performance with theirs.

Note that we do not consider the claim that NC interpretation is influenced by the context (Downing, 1977; Warren, 1978) since our work is carried out out of context. Hence, we make the assumption that there is a unique semantic relation for a given sense collocation.

4.2 `WordNet::Similarity`

`WordNet::Similarity`[5] (Patwardhan et al., 2003) is an open source software package developed at the University of Minnesota which facilitates the measurement of semantic similarity or relatedness between a pair of concepts, or alternatively a pair of words. The system provides six measures of similarity and three measures of relatedness based on the WORDNET 2.1 (Fellbaum, 1998) lexical database. The measures of similarity are based on analysis of the WordNet `isa` hierarchy. For our purposes, we used the WUP method (Wu and Palmer, 1994) based on the findings of Kim and Baldwin (2005) for an NC disambiguation task over the same set of semantic relations.

4.3 Data Collection

We randomly gathered and annotated two sets of NCs: a 200 NC seed set and a 400 NC test set. The seed set is for use in automatically generating NCs. The test set (which is disjoint with the seed set) is for use in examining the utility of the NCs semi-automatically derived by our method. That is, we use the second to test the performance of existing interpretation methods over the NC data we generate. The NCs in both sets were sourced from the Wall Street Journal (WSJ) component of the Penn Treebank. Also, there is no overlap between the two data sets. SENSEVAL-2/3 and SemCor were obvious alternate candidates given the sense annotations, but were unsuitable due to the small number of NC types.

Two human annotators tagged both the semantic relations and word senses for the NCs in the two sets and met to resolve any annotation disagreements (these NCs were included in the experiments). Over the two data sets, the initial inter-annotator agreement for tagging semantic relations was 52.3% and the inter-annotater agreement for tagging word senses was 58.8%.

The corpus filter used to exclude bogus NCs uses NCs extracted from the British National Corpus (BNC) and Reuters Corpus using a full text chunker. From these NCs that were extracted, we took only noun-noun bi-grams adjoined by non-nouns (e.g. ... *the apple pie was* ...) to ensure that they were not part of a larger compound nominal. We additionally measured the entropy of the left and right contexts for each noun-noun type, and filtered out all compounds where either entropy value was < 1.[6] This was done in an attempt to, once again, exclude NCs which were embedded in larger multiword expressions, such as *service department* in *social service department*. The combined total number of unique NC types extracted from the two corpora was 389,400.

5 Evaluation

We conducted two different experiments. The first was to automatically derive NCs using the pro-

[5] `www.d.umn.edu/~tpederse/similarity.html`

[6] For the left token entropy, if the most-probable left context was *the*, *a* or a sentence boundary, the threshold was switched off. Similarly for the right token entropy, if the most-probable right context was a punctuation mark or sentence boundary, the threshold was switched off.

		NewNC (A)	PosNC (B)	SRError in (B)	Filtered (C)	Sample (D)	NegNC in (D)	SRError in (D)	AllError in (D)	Accuracy of (A)
Syn	1	384	69	21.74%	315	63	7.94%	27.59%	33.33%	68.75%
	2	229	13	7.69%	216	43	13.95%	18.92%	30.23%	**69.44%**
Hyp	1	616	100	25.00%	516	103	0.97%	30.39%	30.10%	**70.78%**
	2	997	87	22.99%	910	182	6.04%	40.94%	44.51%	57.37%
	3	1,202	70	40.00%	1,132	226	11.06%	44.28%	50.44%	50.25%
	4	1,079	58	43.10%	1,021	204	18.14%	47.90%	57.35%	43.47%
Sis	1	10,728	686	24.05%	10,042	502	7.97%	30.52%	36.06%	**64.72%**
	2	192,917	1,751	30.26%	191,202	956	14.33%	24.79%	35.56%	64.48%
	3	19,275	106	19.81%	19,169	958	7.31%	36.37%	41.02%	59.11%
	4	44	0	0.00%	44	44	68.18%	100.00%	100.00%	0.00%

ROW LABELS:

Syn = synonyms over 1–2 iterations
Hyp = hypernyms over 1–4 iterations
Sis = sister words over 1–4 iterations

COLUMN LABELS:

NewNC (A) = number of newly derived NCs
PosNC (B) = derived NCs positively attested in the corpus
SRError in (B) = ratio of incorrect NCs in (B)
Filtered (C) = NCs not attested in corpus and filtered out
Sample (D) = test sample (sub-sample of (C))
NegNC in (D) = ratio of non-NCs in (D)
SRError in (D) = ratio of incorrectly tagged NCs in (D)
AllError in (D) = total error ratio in (D)
Accuracy in (A) = ratio of NCs tagged correctly in (A).

Table 2: Results for the three ontological relations

posed method. The second was to interpret NCs using existing interpretation techniques including the derived NCs as training data. For the second experiment, we used the interpretation methods introduced in Moldovan et al. (2004) and Kim and Baldwin (2005).

5.1 Experiment I

In this experiment, we tested our proposed method over the three ontological relation types of synonyms, hypernyms and sister words. There are two points of interest in this experiment. The first is how many NCs we are able to derive and the second is how many of the derived NCs are tagged with the correct semantic relation. We test each of these separately for the three ontological relation types.

The results of experiment one are reported in Table 2, for the three ontological relation types over subsequent iterations for each. Here, *NewNC (A)* is the number of newly generated NCs, *PosNC (B)* is the number of NCs remaining after corpus filtering; *Filtered (C)* is the number of NCs filtered out by the corpus filter (note A = B + C); and *Sample (D)* is a sub-sample of instances from (C) for evaluation purposes; and *Accuracy of (A)* is the proportion of correctly generated NCs out of the original (A). Note that negative NCs (NegNC) are pragmatically or semantically infelicitous NCs.

The evaluation is performed from two perspectives. One is to check whether the generated NCs are positively attested in the corpus data. The other is to check the soundness of the predicted semantic relations in the generated NCs. Two human annotators manually checked the correctness of the semantic relations in the newly generated NCs. To evaluate the plausibility of the generated NCs, we categorized the generated NCs into 2 sets based on their corpus attestation: (B) = NCs attested in the corpora, and (C) = NCs not attested in the corpora. The annotators manually checked all the NC instances found in the corpus and a random subsample of the NCs not found in the corpus (due to the resource constraints).

The human annotators checked an average of 180 NCs per hour. The proportion of NCs in (D) tagged with an incorrect semantic relation was computed after removing the NCs not attested in the corpus.

Table 3 shows the cumulative error over subsequent iterations for each ontological relation type.

5.2 Experiment II

The second experiment was aimed at evaluating the utility of the NCs generated by our proposed method. Ideally, we would have liked to have performed extrinsic evaluation over the generated NCs, e.g. in an Machine Translation context. However, due to time limits, we instead only test the generated NCs as additional training data in an NC interpretation task.

In this experiment, we used the NCs generated

		SRError in (B)	SR+NegNCError in (C)	TotalError in (A)
Synonym	1	21.74%	33.33%	31.25%
	2	19.51%	32.08%	30.56%
Hypernym	1	25.00%	30.10%	29.22%
	2	24.06%	39.65%	37.51%
	3	28.40%	44.42%	57.26%
	4	31.11%	48.11%	42.74%
Sister	1	24.05%	36.06%	35.28%
	2	28.49%	35.73%	35.50%
	3	28.12%	37.83%	35.97%
	4	28.12%	38.94%	35.67%

SRError in (B) = cumulative error ratio in (B)
SR+NegNC Error in (C) = cumulative error ratio in (C)
TotalError in (A) = ratio of NCs tagged correctly in (A)

Table 3: Cumulative error in Experiment I

Semantics	Correct	All	ErrorInData2
Synonym	338	813	22.88%
Hypernym	668	3,015	39.90%
Sister	1,042	10,928	34.64%
Combined	1,648	14,356	32.47%

Table 4: Data set sizes in Experiment II

Semantics	Method	seedNC	Correct	All
Baseline	0R	23.00%	–	–
	M+	33.25%	–	–
	K+B	29.75%	–	–
Synonym	M+	–	33.25%	32.50%
	K+B	–	29.00%	29.50%
Hypernym	M+	–	32.50%	32.75%
	K+B	–	**30.50**%	29.00%
Sister	M+	–	**35.50**%	34.25%
	K+B	–	29.00%	28.00%
Combined	M+	–	34.75%	34.00%
	K+B	–	29.75%	29.00%

Table 5: Results of NC interpretation (0R = Zero-R majority vote, M+ = Moldovan et al. (2004), and K+B = Kim and Baldwin (2005))

Semantics	Instances (ErrorRate)	M+	K+B
Manual	1769 (0.00%)	–	42.00%
Synonym	613 (30.34%)	32.50%	29.50%
Hypernym	616 (29.32%)	31.75%	**32.75%**
Sister	686 (24.05%)	29.00%	28.00%

Table 6: Results of interpretation with a fixed amount of training data, with and without noise in the data

by our method as training data within existing interpretation methods. In this, we are interested in: (1) how well the methods operate using the generated NCs as training data relative to the size and nature of data; and (2) given similar amounts of training data and error rates, whether data derived from the different ontological relation types leads to different interpretation results. In this, we use two previous methods: Moldovan et al. (2004) and Kim and Baldwin (2005).

For the first test, we used two data sets: (1) all *correct* data (including seed NCs) derived using the three ontological relation types, i.e. all NC instances that were manually verified to be correct (*Correct*); and (2) all generated NCs from (B) (including the seed NCs once again) irrespective of correctness (*All*). In the second instance, we took all NCs generated over all iterations from synonyms and hypernyms. Since the number of NCs derived using sister words were large, we decided to use only the ones generated in the first iteration. The data set sizes are shown in Table 4. We use a zero-R majority class method as the baseline for the interpretation task. [7]

[7] The majority of instances in the test set have the TOPIC semantic relation

6 Discussion

In the first experiment, we predictably derived the largest number of NCs via sister words. After the first iteration, from the 200 seed NCs, we generated 384, 616 and 10,728 NCs using synonyms, hypernyms and sister words, respectively. Even when we used only the positive NCs found in the corpus (i.e. strict attestation filtering), the numbers of newly generated NCs are 69, 100, and 686, respectively. As such, we can see that our method is successful in automatically deriving a large number of NCs.

When we looked at the error rate of the newly derived NCs, we observed that the NCs generated via synonyms contain less errors than the other two ontological relations. However, from an interpretation point of view, the accuracies in using hypernym and sister word are relatively high compared to accuracies reported for previous methods, namely 43.2% for M+ (Moldovan et al., 2004) and 53% for K+B (Kim and Baldwin, 2006), as compared to between 64.7% and 70.8% after the first iteration of our method. In addition, our method needs only a small number of manually tagged NCs (200 NCs in this experiment) compared to prior approaches. As a result, we strongly believe that our method can efficiently derive NCs tagged with semantic relations with a lower error rate and reduced human effort. Also, our method has the potential to significantly increase the volume of automatically interpreted NCs.

Note that when we use hypernyms and sister words we go beyond the scope of the original semantic collocation, so that we can significantly increase the number of tagged NCs by iterating repeatedly. In Table 2, we can see that the increase in error rate as we get further and further away from the original semantic collocations is gradual (linear). Also, in the filtered (unattested) NCs, the error increases gradually, and that in excluding these in our method, the error rate is kept down, especially with synonyms and sister words. Considering that the number of generated NCs is exponential, especially with hypernyms and sister words, the total error rate is relatively low. Finally, the results in Table 3 show that the error rate for NCs derived via synonyms and sister words is linear even after several iterations. However, we believe that although we used two filtering methods for reducing noise in the derived data sets, we require more efficient methods to effectively avoid noise in generated data. We leave this as an area for future work.

The second experiment in Table 5 was to test the utility of the generated NCs for existing interpretation methods. Although the number of NCs generated by our method is large, the range of sense collocations we capture is restricted, especially for synonyms. The baseline for each method was to use only the 200 seed NCs as training data, which in itself outperformed a simple majority class baseline for both methods tested. Using additional training data generated by our method, we were able to improve over this baseline performance. Although the number of generated NCs is 5 times more than the 200 seed NCs, the performance of Moldovan et al. (2004) does not increase significantly. This confirms that idea that Moldovan et al. (2004) relies not only on the amount of training data but also on the range of sense collocations in the training data.

We also analyzed the performance of the two methods using error-free (Correct) and "error-full" (All) training data. Despite the smaller amount of training data, the best performance is achieved using the Correct data in Moldovan et al. (2004). This is disappointing in terms of the general-purpose NC interpretation task targeted in this paper, but very promising for domain-specific NC interpretation. Our reasoning here is that the filtering could be tailored to a given domain, such that we could generate NCs with higher lexical similarity to NCs in that domain, and hence a higher chance of producing a correct classification.

Finally, we evaluated the performance of interpretation methods in terms of the three different ontological relations over similar amounts of training data. In Table 6, the manual performance uses the 1769 hand-tagged training instances. We were unable to tag word senses for all of the training data. As a result, we have shown the performance for only Kim and Baldwin (2005). Using the three different ontological relations, we constructed similar amounts of training data. All the training data included only corpus-attested NCs with correctly tagged semantic relations. Although we achieved the best performance from hypernym semantics using Kim and Baldwin (2005), the difference in performance is not significant. As a result, we conclude that there is no significant difference between the three relation types with respect to the task of NC interpretation.

7 Conclusion

In this paper, we proposed a novel method for automatically deriving NCs tagged with semantic relations using bootstrapping and sense collocations. We used synonyms, hypernyms and sister words as the ontological relations for the substitution of constituent in NCs. We were able to achieve between 64.72% and 70.78% accuracy in deriving NCs and we found that the error rate while deriving NCs increased nearly linearly when using synonyms and sister words. We also demonstrated the utility of the derived NCs as training data, and concluded that the performance of interpreting NCs depends not only on the amount of training data but also the sense collocations in the training data. The main concern of this approach is to interpret NCs within the same or similar scope of sense collocation. As future work, we will further exam the proposed method to expand not only interpreted NCs but also the variation of sense collocations.

References

Ken Barker and Stan Szpakowicz. 1998. Semi-automatic recognition of noun modifier relationships. In *In Proceedings of the 17th international conference on Computational linguistics*, pages 96–102.

Pamela Downing. 1977. On the creation and use of English compound nouns. *Language*, 53(4):810–842.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, USA.

Timothy Wilking Finin. 1980. *The semantic interpretation of compound nominals*. Ph.D. thesis, University of Illinois, Urbana, Illinois, USA.

Karen Sparck Jones. 1983. Compound noun interpretation problems.

Su Nam Kim and Timothy Baldwin. 2005. Automatic interpretation of compound nouns using wordnet::similarity. In *In Proceedings of 2nd International Joint Conference on Natual Language Processing*, pages 945–956. Jeju, Korea.

Su Nam Kim and Timothy Baldwin. 2006. Interpreting semantic relations in noun compounds via verb semantics. In *In Proceedings of COLING/ACL*, pages 491–498. Sydney, Australia.

Mark Lauer. 1995. *Designing Statistical Language Learners: Experiments on Noun Compounds*. Ph.D. thesis, Macquarie University.

Judith Levi. 1979. The syntax and semantics of complex nominals. In *New York:Academic Press*.

Dan Moldovan, Adriana Badulescu, Marta Tatu, Daniel Antohe, and Roxana Girju. 2004. Models for the semantic classification of noun phrases. In *HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, pages 60–67.

Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. 2003. Using measures of semantic relatedness for word sense disambiguation. In *In Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*.

Lucy Vanderwende. 1994. Algorithm for automatic interpretation of noun sequences. In *In Proceedings of the 15th conference on Computational linguistics*, pages 782–788.

Beatrice Warren. 1978. *Semantic Patterns of Noun-Noun Compounds*. Ph.D. thesis, Actr Universitatis Gothoburgensis.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 133 –138.