“Nut Case: What does It Mean?”: Understanding Semantic Relationship between Nouns in Noun Compounds through Paraphrasing and Ranking the Paraphrases

Derry Tanti Wijaya
Language Technologies Institute
Carnegie Mellon University
5000 Forbes Ave
+14122688525
dwijaya@cs.cmu.edu

Philip Gianfortoni
Language Technologies Institute
Carnegie Mellon University
5000 Forbes Ave
+14122688525
pwg@cs.cmu.edu

ABSTRACT
A noun compound (NC) is a sequence of two or more nouns (entities) acting as a single noun entity that encodes implicit semantic relation between its noun constituents. Given an NC such as 'headache pills' and possible paraphrases such as: 'pills that in-duce headache' or 'pills that relieve head-ache' can we learn to choose which verb: 'induce' or 'relieve' that best describes the semantic relation encoded in 'headache pills'? In this paper, we describe our approaches to rank human-proposed paraphrasing verbs of NCs. Our contribution is a novel approach that uses two-step process of clustering similar NCs and then labeling the best paraphrasing verb as the most prototypical verb in the cluster. The approach performs the best with an average Spearman’s rank correlation of 0.55. This approach, while being computationally simpler, gives a better ranking than the current state of the art. The result shows the potential of our approach for finding implicit relations between entities especially when the relations are not explicit in the context in which the entities appear, rather they are implicit in the relationship between its constituents.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *clustering, retrieval models, search process, selection process.* I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *language models, text analysis.*

General Terms: Algorithms, Measurement, Performance, Experimentation, Languages.

Keywords: Noun compounds, semantic relation, clustering, paraphrasing, ranking.

1. INTRODUCTION
Noun compounds (NCs) such as 'apple cake' or 'afternoon rain' are sequences of two or more nouns acting as single nouns. They encode implicit semantic relations between the nouns they contain. For example, the NC 'nut bread' concisely encodes the relation 'X that contains Y' as in 'bread that contains nut'. In this case, 'contain' is the paraphrasing verb that best describes the semantic relation encoded in the NC 'nut bread'. Interpreting semantic relations encoded within NCs is an important goal for broad-coverage semantic processing due to the high frequency of occurrence of the NCs and their high productivity in English [1]. The task of interpreting semantic relation however, is challenging because although the NCs are very common in English (due to their high productivity), their frequency distribution follows a Zipfian law in which the majority of NCs encountered is of rare types. Furthermore, semantic relation encoded within the head and modifier nouns in an NC is implicit and influenced by contextual and pragmatic factors [2].

However challenging the task is, understanding semantic relations encoded within NCs are important, especially for many natural language applications such as question answering, machine translation, web search, or micro-reading. One way to interpret them is to classify them under abstract relations such as cause, container, source, time, and location. Paraphrasing is another way to interpret NCs that are directly usable by NLP applications. For example, a web search engine can use suitable paraphrasing of 'headache pills' as 'pills that relieve headaches' and 'sleeping pills' as 'pills that induce sleeping' to refine the query and return better ranking of search results. Such paraphrasing can also benefit machine translation system that translates NCs from English to other languages. In information extraction, paraphrasing NCs such as 'WTO Geneva headquarters' as 'Geneva headquarters of the WTO' or as 'WTO head-quarters located in Geneva' might help in co-reference resolution [1]. Furthermore, by paraphrasing NCs, fine-grained semantic relations between the nouns can be discovered; which may provide richer information content of a document useful for micro reading.

In this paper, we describe our approach for ranking paraphrasing verbs of NCs for Task 9 of the SemEval-2 workshop [3]. In this task, each NC has a list of human-proposed paraphrasing verbs and the number of annotators who proposed that paraphrase. An extract from the training set:

flu virus	cause	38
flu virus	spread	13
…		
flu virus	be made up of	1
flu virus	exacerbate	1

SMER’11, October 28, 2011, Glasgow, Scotland, UK.

The test set has similar format, though the frequency (the number of annotators who proposed the paraphrase) is not included and the paraphrases appear in random order:

…
chest pain originate
chest pain descend in
chest pain be in
…

For each NC, the gold standard ranking of its paraphrases is the ranking of the verbs based on the number of annotators that propose them. The task is to provide for each NC in the test set, a ranking of its verbs that is as close as possible to the gold standard ranking. In the next section, we describe related works for this task and our proposed method in section 3. We describe our experiments in section 4 and conclude in section 5.

2. RELATED WORKS

The current state of the art system for this task uses classifier to determine for each verb pair describing a noun compound which verb should be ranked higher [4]. The classifications are combined into one ranking by assigning one point to a verb each time it ranks higher than another verb for that NC. The sums of the points are then used to rank the verbs for each NC. The system uses internal features: the distribution information of the verbs in the training data (e.g. how many times verb1 is ranked higher than verb2) and external features: the semantic relations between the verb and the NC as computed from WordNet and the co-occurrence frequencies count of the sequence *noun2 verb noun1* in Google N-gram corpus. The system uses memory-based learning for classification. Although this system is currently the best for the task, it is computationally expensive. In order to train the model it uses all possible verb pairs in the training data as training instances, amounting to more than one million training instances. Furthermore, before it can generate the ranking of the verbs, it needs to do classification on all verb pairs in the test data, amounting to almost three million test instances to classify. In their experiment, they discover that the best feature to use is the simple distribution counts of the verbs in the training data: how many times verb1 is ranked higher than verb2.

The second best system for the task also uses the simple distribution counts of the verbs in training data to rank the verbs in test data [5]. They start with the assumption that people tend to use more general, semantically light paraphrases than detailed, semantically heavy ones. Hence, the list of verbs proposed for an NC must have indicated the same interpretation, varying only in the degree of semantic details (with general verbs ranking higher than detailed verbs). The system simply ranks more prototypical, general, frequently co-occurring verbs higher than detailed verbs for each NC. No semantic information about NCs was used, nor was the frequency provided in the training set used: it only uses 0 or 1 as values (i.e. either verb1 has co-occurred with verb2 in the same NC or not). To compute generality of a verb, the system first generates conditional probability table for each verb occurring; given that another verb has co-occurred with it in the same compound in the training data: i.e. P(verb1|verb2) = P(verb1 and verb2 occur in the same compound) / P(verb2). Then, for a test NC with a list of verbs to rank, the score of a verb in the list is computed as the sum of its conditional probability with every other verb in the list. The higher the sum, the higher is its generality and rank. Although the system only uses a very simple ranking scheme, it is the second best system for ranking candidate paraphrases, highlighting the importance of the distribution counts of verbs in training data.

However, just like the best system, the second best system requires the computation of conditional probabilities for all verb pairs in the training data, amounting to more than one million verb pairs. Also, since it prefers general verb and does not take into account the frequencies provided in the training data, it often rank a verb highly by virtue of its generality. For example, for 'bathing suit', the highest ranked verbs are 'be in', 'be found in' which do not make sense for this NC but ranked high because they are general verbs that co-occur frequently with other verbs.

Since both best systems [4] and [5] discover verbs distributions feature to be important, our approach exploits verbs distributional similarity to compute *k* most similar NCs for each test NC. Indeed, if we think of the verbs proposed by human annotators for an NC as its context, using this context, we can conjecture that two NCs have similar relationship between its noun constituents if they share similar verbs distributions. We then perform 'local' ranking of verbs based only on these neighboring NCs' verbs distributions. Our approach improves the current state of the art performance despite being very simple in its approach, feature and computational complexity.

3. PROPOSED METHOD

Our approach is based on the conjecture that similar NCs have similar paraphrasing verbs distributions. This is because similar verbs will tend to be co-proposed for similar NCs by human annotators. Hence, a good verb for describing one NC (one proposed many times for the NC) may be good for describing other similar NCs. In a kind of topic-modeling approach where documents and words are linked by latent variables that are topics; here we link NCs and verbs by latent 'topics' which are clusters of similar NCs.

This idea is similar in spirits to the idea of distributional clustering [6] that clusters words according to the distributions of contexts in which they appear. However, in this task we cluster noun compounds according to the context-independent distributions of the verbs proposed for these NCs instead of their context-dependent distributions in documents. This is because, as we will see in section 4, the relationships between nouns in noun compounds are implicit, they are seldom explicitly mentioned in documents where they appear. Previous works in noun compound's interpretation have shown that measuring relation between NCs by similarity of contexts: words in sentences in which both noun constituents appear does not perform better than simply using word (noun constituent's) similarity [7]. These findings suggest that implicit relations have to be represented indirectly, either through noun constituents' similarity as in [7] or through verbs distributional similarity as in our approach. Thus, though similar in spirits to the idea of distributional clustering in [6], the words and their distributions that we are interested in using are different.

Given a test NC and its list of human-proposed verbs that we need to rank, we first find *k* NCs that are most similar to this test NC based on the verbs distributional similarity (clustering-step). Then we score and rank each of the verbs of this test NC based on how much the verb co-occurs with the majority of NCs in this cluster of *k* NCs (labeling-step). This cluster-then-label approach uses only co-occurrence counts of the NCs and the verbs. Our proposed cluster-then-label approach is similar to [5] in that we use verbs distributions counts as feature. However, we differ in several aspects. Although in [5], they recognize that a list of verbs proposed for an NC must have indicated the same interpretation of the NC, they fall short of using this information as a feature and focus instead on the generality of the verbs. We, on the other hand, believe that verbs distributions information is an important contextual features for finding other, similarly interpretable NCs. Also, ranking in [5] is conducted globally by computing generality of a verb with respect to all other verbs in the data. In our approach, ranking is done locally with respect to only the verbs in the nearest *k* NCs.

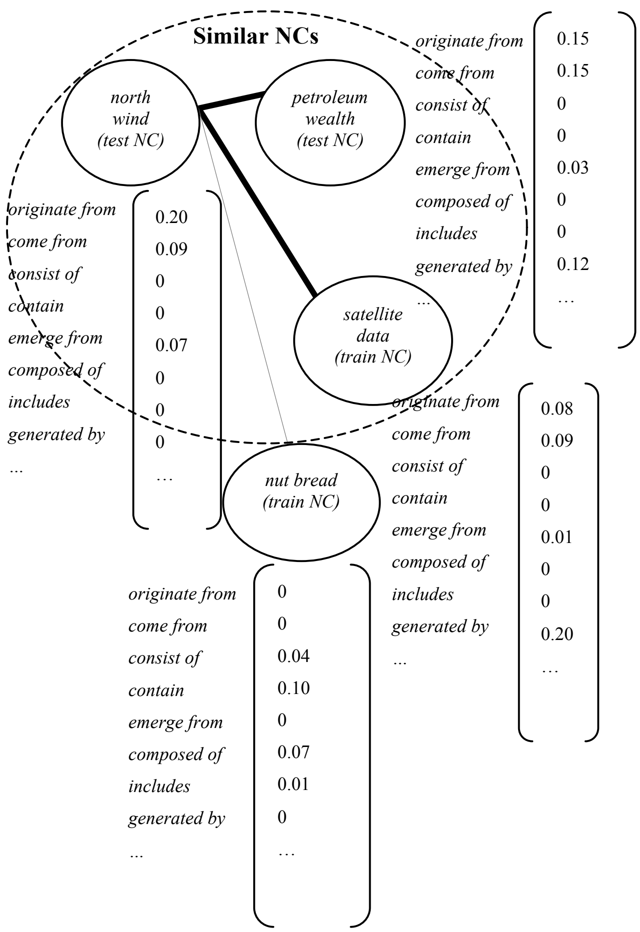

Figure 1. Grouping Similar NCs

To compute *k* most similar NCs for each test NC, we construct a feature vector for each NC. Feature vectors for train NCs contain tfidf values of verbs in the data set. Idf values of verbs are computed by treating NCs as documents. Document frequency (df) of a verb is therefore the number of NCs that have this verb proposed by a human annotator. For test NCs, we do not have information on the number of human annotators. The only information we have is what verbs are candidate phrases for the test NC that we need to rank. Hence, for a test NC we only use binary counts: 1 (if the verb is candidate phrase for the NC) or 0 (if the verb is not candidate phrase for the NC) for co-occurrence counts and for computing tfidf values. Similarities between NCs are then computed as cosine similarities between their feature vectors. In figure 1 we illustrate this clustering step. Edges are weighted by cosine similarity, the more similar the NCs, the heavier (bolder) the edge.

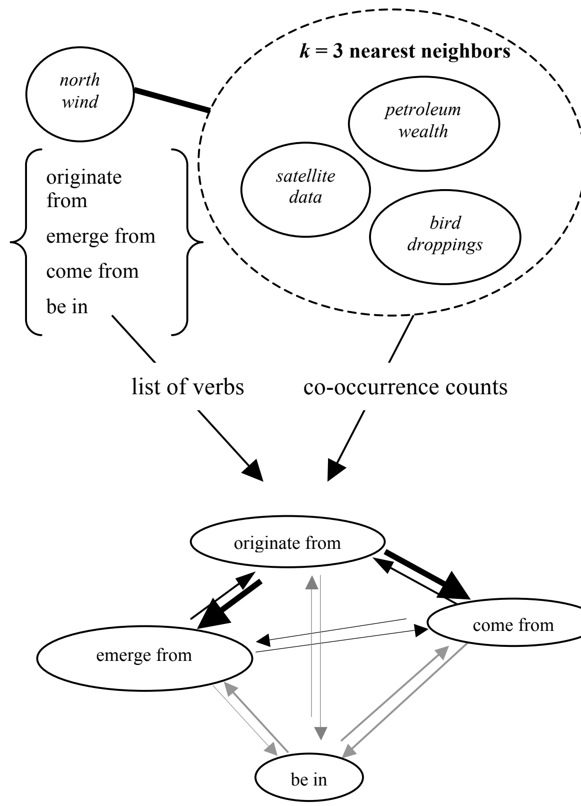

Figure 2. Labeling Verbs: Directed Edge Between Verbs, e_{ij} is weighted by P(verb$_i$|verb$_j$)

Then, in the labeling step, for each test NC, we rank its candidate verbs based on which verb is most frequently co-occurring in its group of *k* NCs. To do so, we compute the score of each verb v as the sum of its 'pseudo' conditional probability (i.e. relative co-occurrence frequency) with every other verb v_i for that NC. Our ranking method is similar to [5], but computed locally, only on the co-occurrence counts of the verbs in the *k*-nearest NCs (Figure 2).

$$score(v) = \sum_{v_i \neq v \in NC} P(v|\, v_i)$$

$$P(v|\, v_i) = \frac{\sum_{j=1}^{k} \min(\# v, \# v_i) \ in \ NC_j}{\# \ total \ occurrence \ of \ v_i \ in \ all \ k - nearest \ NCs}$$

To compute co-occurrence counts of the verbs in *k*-nearest NCs, we use frequencies provided in the training data. If verb v occurs n_1 times in an NC and verb v_i occurs n_2 times in the same NC, then their co-occurrence count for that NC is computed as the minimum of n_1 and n_2, i.e. the overlap of their occurrences. In practice, we can experiment with other ways of scoring (labeling) the verb; for example, by using PageRank [8] on the matrix of verbs' co-occurrences to rank the verbs based on popularity.

4. EXPERIMENTS

4.1 Setup

For experiments, we train and test all our approaches on data set provided in [3]. The training data consists of 250 NCs, each paraphrased by 25-30 human annotators. The test data contains 338 NCs, each paraphrased by 50-100 human annotators. For each NC in the test set, we provide a ranking of its verbs. We compute the score of our ranking for each test NC using the official evaluation measure provided in [3], a variant of Spearman's rank correlation coefficient that allows for tied ranks:

$$\rho = \frac{n\sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

where n is the number of verbs to rank for the NC, X and Y are human and system-proposed rankings respectively with fractional ranks (average of the positions of the duplicated ranks) assigned in case of duplicates. The final score is the average of the scores over test NCs.

4.2 Results and Discussion

Our experiment results are summarized in Table 1 where our systems are shown in bold. The baseline involves scoring a given verb by simply its frequency (i.e. its prior) in the train set. We experiment with different values of k = 5, 10, 15, 20, 30 and note that the performance of our approach increases with k and stabilizes at k = 15.

The performances and descriptions of all other systems and the baseline system in Table 1 are reported in [9].

Our simple approach that uses only verbs distributions counts and co-occurrences with NCs in the data set, gives the best result that is significantly better than the result of the current best system (UvT) [4]. Our approach is also computationally simpler. For a test set of n NCs and v verbs (where $v \gg n$), UvT at the worst case necessitates $O(v^2)$ classifications, while our cluster-then-label approach at the worst case only necessitates $O(n^2)$ to compute similarities between NCs.

Furthermore, our approach uses only simple features. In [7], contextual features to model semantic similarity between NCs are categorized into three classes of context types: word similarity, relation similarity between noun constituents of the NCs, and type similarity between NCs. Our approach does not use any word similarity derived from WordNet or relation similarity derived from web data like what is used in UvT and in [7] or [10]. Instead, our approach uses verbs distributions similarity to measure relation similarity between NCs. This is important for this task because although many NCs in this task do not share common nouns (i.e. no word similarity) or similar contexts in sentences or documents (i.e. no type similarity); in this task they share similar verbs distributions that indicate relation similarity between their nouns.

For example, 'north wind', 'satellite data', and 'bird droppings' are different NCs that do not share common nouns (word similarity) or appear in similar contexts in sentences or documents (type similarity). It is unlikely, for example, that we will see sentences or documents where 'satellite data' and 'bird droppings' are used in similar contexts. This form of type similarity has also other conceptual problem in which other NCs such as 'weather channel' are likely to appear in similar contexts with 'satellite data' even though they encode different implicit relations.

However, these NCs: 'north wind', 'satellite data', and 'bird droppings' do share similar relation in that the head nouns: 'wind', 'data', and 'droppings' might 'come from' (or 'originate from' or 'emerge from') but definitely not 'made up of' (or 'consist of' or 'contains' or 'includes') the modifier nouns: 'north', 'satellite', and 'bird'. Hence, if we know that the set of candidate verbs for 'satellite data' is similar to the set of candidate verbs for 'north wind', knowing that the verb 'originate from' is the best to describe the relation between the noun constituents of 'satellite data' could suggest that it is also the best verb to describe the relationship between the noun constituents of 'north wind'. Based on this assumption, we expect that for an NC that have many other similar NCs (many NCs with similar verbs distributions), their ranking of verbs would be better informed than an NC with few other similar NCs.

Analysis of the NCs for which our approach performs the best and the worst supports our expectation. For NCs such as 'sea mammals' that has many candidate verbs ('live in', 'be found in', 'come from', 'be in', 'reside in') co-occurring with other NCs: 'sea animals', 'sea urchins', 'field mouse', 'desert rat', 'city folk', 'kennel puppies'; our approach is able to output a good ranking of its verbs (average Spearman ρ of 0.77). However, for NC such as 'university education' that does not have any co-occurring verbs except for a general verb like 'come from' with other NCs, has a worse ranking of its verbs (average Spearman ρ of 0.18).

There are at least two possible explanations why we believe this can happen. Firstly, it can be that we do not have enough similar NCs in our data set to help the system interpret the semantic relation encoded within 'university education'. Secondly, it can be that 'university education' is a very general, ambiguous NC with no clear semantic interpretation. An investigation into the list of human-proposed verbs for 'university education' suggests that the first explanation maybe the answer. 'university education' has top-ranked verbs such as 'come from', 'be in', 'be given in', 'be provided by', 'be given by'. No other NCs in our train or test sets have these verbs proposed together in the same NC. This suggests the lack of NCs in our data set that encodes similar relation to 'university education'. This results in the poor ranking of this NC's verbs: there are simply not enough other NCs with similar semantic relation to help interpret this NC.

The fact that our approach gives better ranking than UvT, even when relation between a verb and an NC is measured on a much larger data in UvT also suggests an important point that implicit relation encoded in an NC may not always be directly expressible in a sentence, for example in the form of the sequence: *noun2 verb noun1*. Since the relation is implicit, such sequence maybe hardly ever found in sentences. For example, sentences will not normally state the sequence 'bread that contains nut' when they can simply state 'nut bread' to refer to the same concept. Measuring relation between NCs by similarity of contexts: words in sentences in which both noun constituents appear also do not perform better than simply using word (noun constituent's) similarity, perhaps for the same reason that type similarity is conceptually problematic [7]. These findings suggest that implicit relations have to be represented indirectly, either through noun constituents' similarity as in [7] or through verbs distributional similarity as in our approach. In our approach, relation is represented indirectly in the set of verbs that co-occur with similar NCs. The cluster of NCs and the verbs that co-occur with them constitutes the 'topic' model that represents the relation.

System	Average Spearman ρ
Cluster-then-label (k=15)	**0.546**
Cluster-then-label (k=10)	**0.546**
Cluster-then-label (k = 20)	**0.543**
Cluster-then-label (k = 5)	**0.536**
Cluster-then-label (k = 30)	**0.535**
UvT-MEPHISTO [4]	0.450
UCD-PN [5]	0.441
UCD-GOGGLE-III	0.432
UCD-GOGGLE-II	0.418
UCD-GOGGLE-I	0.380
UCAM	0.267
NC-INTERP	0.186
Baseline	**0.425**

Table 1: Result of Experiments

By showing improved verbs ranking in a computationally simpler approach, we show that our approach improves scalability without trading off precision in ranking. In fact, both better ranking and faster performance are achieved by our approach compared to state-of-the-art approaches for this task.

5. CONCLUSION AND FUTURE WORKS

In this paper, we have proposed a novel approach for ranking candidate paraphrasing verbs of noun compounds. Our approach for ranking candidate verbs consists of two steps: the clustering of similarly interpretable NCs based on their verbs distributional similarities and the ranking of the verbs based on their co-occurrence counts in the cluster. We conduct experiments and show that our approach, despite being simpler in features and computational complexity, gives significantly better ranking results than the current best system for the task.

However, this task is just one element in the pipeline of a more general paraphrasing approach to noun compound interpretation. In this task, candidate paraphrases for each test NC are assumed to be known (in fact, they are supplied by human annotators). However, such assumption is not true in general. In general, given a test NC, we have to find a set of candidate paraphrases for the NC before we can use our approach for ranking the paraphrases. This is one challenge that we have begun to explore in this paper.

Similar to our clustering step, given a test NC, to automatically propose candidate paraphrases for the NC, we first find *k* NCs from our training data that are most similar to the test NC. Motivated by the findings in [7], [10] and [11] that shows effectiveness of word-level similarity for noun compound interpretation, we compute similarities between NCs as the sum of their noun constituents' similarities as measured in WordNet. We construct a feature vector for each noun containing tfidf values of hypernyms that exist from the nouns in our data set to the root node in WordNet. The similarity between two nouns is then measured as the cosine similarity between their feature vectors.

Once we find *k* similar train NCs, we select candidate paraphrases for the test NC from the pool of verbs constructed from the candidate verbs of the *k* NCs. Similar to our labeling step, we use our method for ranking to select candidate paraphrases from this pool (i.e. based on which verb is most frequently co-occurring in this pool). We select as many candidate paraphrases for the NC as there are candidate paraphrases for the NC in our test data.

k	Average precision	Average Spearman ρ
1	0.21	0.25
5	0.31	0.33
10	0.36	0.35
15	0.37	0.35
20	0.38	0.35

Table 2: Result of Experiments for the More General Approach of Ranking

At different values of *k*, we measure the average precision of our automatically proposed candidates and the average Spearman's rank correlation score over our test data (table 2). The precision of our proposed candidates for an NC is measured as the ratio of the overlap between our automatically proposed candidates for the NC and the human proposed candidates for the NC. For each NC, we also compute Spearman's rank correlation score on candidates that overlap.

Similar to our previous result, the Spearman scores of this more general approach for ranking paraphrases increases with *k* and stabilizes at $k = 15$. Although the Spearman's scores at different values of *k* are lower here than our previous result, this is a result of a more general approach that does not assume human proposed candidates for test NCs. Rather, this approach will be able to automatically find candidate verbs and then rank candidate verbs for any test NC. In the future we will explore more measures and/or strategies to find better *k* similar NCs and to propose better candidate paraphrases.

We also observe from the results of our experiments that a noun compound with more similarly interpretable NCs in our data have better ranking of its paraphrases than a noun compound with few similarly interpretable NCs. Given a noun compound, can we enrich our data by gathering more similarly interpretable NCs? One possible way to achieve this is by expanding each noun constituent in the NC with their hyponyms or sister terms from WordNet. By trying out different combinations of these expanded sets of modifier nouns and head nouns and finding valid noun compounds through web search, for example, more similarly interpretable NCs can be found.

Considering that our approach can be represented in a graph where nodes are NCs and verbs, and there are edges between NCs and their paraphrase verbs and edges between NCs that are similar; an interesting further direction is to use various graph-based semi-supervised learning such as [12] to infer the confidence that a noun compound can be paraphrased by a verb. Our initial experiment with [12] is encouraging, resulting in an average Spearman's rank correlation score of 0.51 with default setting of the parameters. We are exploring other ways to improve ranking with this graph-based algorithm, one obvious way is to fine tune the parameters of the algorithm via cross validation, for example.

One limitation of this task is that the ranking of verbs proposed by human annotators are context independent while in reality, this ranking of verbs is very much context-dependent. For example, in some contexts, 'be in' verb for the noun compound 'chest pain' may rank higher in documents that talk about the *current* location of the pain, while the verb 'originate from' for 'chest pain' may rank higher in documents that talk about the origin of the pain (the current location of the pain itself may have already changed). In future it will be interesting to use both the context-independent proposed verbs distributions and the context-dependent words distributions in documents where the NCs appear to inform a better context-dependent interpretation of the noun compounds.

Lastly, given that our experimental results seem to add to and confirm the findings in [7] that co-occurrences, distributional similarity, and noun similarity are important features for modeling and extracting implicit relations between noun constituents in NCs, an interesting future direction for us is to explore the application of the approaches in NC interpretation to model and extract other implicit relations between entities in real world.

6. ACKNOWLEDGMENT

We would like to express our thanks to Partha Pratim Talukdar for his assistance with Junto toolkit [13]. This work was supported in part by DARPA (under contract numbers FA8750-08-1-0009 and AF8750-09-C-0179), by Google and by Fulbright. Any opinions, findings, conclusions and recommendations expressed in this paper are the authors' and do not necessarily reflect those of the sponsors. Finally, we thank the anonymous reviewers for their helpful comments.

7. REFERENCES

[1] Butnariu, C. et. al. 2009. SemEval-2010 Task 9: The Interpretation of Noun Compounds Using Paraphrasing Verbs and Prepositions. *In Proceedings of the NAACL Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW-2009) at NAACL*, pages 100-105.

[2] Lapata, M. 2002. The disambiguation of nominalizations. *Computational Linguistics 28*, pages 357-388.

[3] Butnariu, C. et. al. 2008. SemEval-2010 Task 9: Noun Compound Interpretation Using Paraphrasing Verbs and Prepositions. 29 June 2011. http://docs.google.com/View?docid=dfvxd49s_35hkprbcpt.

[4] Wubben, S. 2010. UvT: Memory-Based Pairwise Ranking of Paraphrasing Verbs. *In Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 260-263.

[5] Nulty, P. and Costello, F. 2010. UCD-PN: Selecting General Paraphrases Using Conditional Probability. *In Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 234-237.

[6] Pereira, F. et. al. 1993. Distributional Clustering of English Words. *In Proceedings of the 31st Annual Meeting on Association for Computational Linguistics*, pages 183-190.

[7] Ó Séaghdha, D. and Copestake, A. 2007. Co-occurrence Contexts for Noun Compound Interpretation. *In Proceedings of the ACL-07 Workshop on A Broader Perspective on Multiword Expressions (MWE-07)*, pages 57–64.

[8] Brin, S. and Page, L. 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *In Proceedings of the 7th International World Wide Web Conference*.

[9] Butnariu, C. et. al. 2010. SemEval-2 Task 9: The Interpretation of Noun Compounds Using Paraphrasing Verbs and Prepositions. *In Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 39-44.

[10] Kim, S. N. and Baldwin, T. 2005. Automatic interpretation of noun compounds using WordNet similarity. *In Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP-05)*, pages 945–956.

[11] Ó Séaghdha, D. 2007. Annotating and learning compound noun semantics. *In Proceedings of the ACL-07 Student Research Workshop*.

[12] Talukdar, P. and Crammer, K. 2009. New regularized algorithms for transductive learning. *European Conference on Machine Learning (ECML-09)*.

[13] Talukdar, P. and Baldridge, J. 2011. Junto: The Label Propagation Toolkit. 29 June 2011. http://code.google.com/p/junto/.