A Knowledge-Based Approach to Syntactic Disambiguation of Biomedical Noun Compounds

Ramakanth KAVULURU and Daniel HARRIS
Division of Biomedical Informatics, University of Kentucky, Lexington, KY, USA
{ramakanth.kavuluru, daniel.harris}@uky.edu

Abstract

Noun compounds (NCs) provide a convenient way of communicating complex biomedical concepts in natural language. New NCs evolve with scientific progress in various fields and are often not included in standard dictionaries. Thus, semantic analysis of NCs is an important task in applications including ontology alignment, semantic data integration, information extraction, and question answering. A first step in such analysis is the syntactic grouping or bracketing of the constituent nouns. The state-of-the-art in bracketing is mostly limited to compounds with three nouns using empirical studies involving corpora like the Web or Medline biomedical research article citations. Here, we present an alternative knowledge-based approach using the Unified Medical Language System (UMLS) concept labels and definitions for NCs with three or four tokens. Experiments indicate that our method offers comparable accuracy with those that use the Web or Medline for 3-token NCs. Preliminary evaluations with 4-token NCs also point to the potential of our approach to bracketing longer NCs.

Keywords: noun compounds, bracketing, terminologies, knowledge-based methods.

Translation in **Telugu**

Title: జీవ-వైద్య శాస్త్రాలలో ఎదురయ్యే ఆంగ్ల సమ్మేళన నామవాచక వాక్య-నిర్మాణ సందిగ్ధతను తొలగించే ఒక శాస్త్ర-జ్ఞానాధారిత విధానం

Authors: రమాకాంత్ కవులూరు, డేనియల్ హారిస్

Abstract: ఆంగ్లములో క్లిష్టమైన జీవ, వైద్య శాస్త్ర భావనలను సహజ భాషలో వ్యక్తపరచుటకు సమ్మేళన నామవాచకాలు ఒక అనుకూలమైన మార్గాన్ని అందిస్తాయి. శాస్త్రీయ ప్రగతితో పరిణమించే క్రొత్త సమ్మేళన పదాలు ప్రామాణిక నిఘంటువుల్లో సాధారణంగా చేర్చబడవు. అందువలన, జ్ఞాన సంపుటి సంకరణం, జ్ఞానానుసంధానం, జ్ఞాన సంగ్రహణం, మరియు సందేహ నివృత్తి వంటి అనువర్తనాలలో సమ్మేళన వాక్యార్థ విశ్లేషణ ఒక ముఖ్యమైన దశ. అటువంటి విశ్లేషణలో సమ్మేళనం లోని అనుసంధాన నామవాచకాల నిర్మాణ వర్గీకరణ ఒక మొదటి పని. ఆధునిక వర్గీకరణ విధానాలు ఎక్కువగా వెబ్ మరియు వ్యాస సారాంశాల పై అనుభావిక అధ్యయనాల ద్వారా త్రిపద సమ్మేళనాలకే వర్తిస్తాయి. ఈ వ్యాసం లో మేము జీవ-వైద్య పరిభాష సంపుటి - యూనిఫైడ్ మెడికల్ లాంగ్వేజ్ సిస్టం - లో ఉండే భావనల పేర్లు మరియు వాటి నిర్వచనాలను ఉపయోగించి త్రిపద-చతుర్పద సమ్మేళనాలను వర్గీకరించే ఒక జ్ఞానాధారిత ప్రత్యామ్నాయ విధానాన్ని ప్రవేశపెడుతున్నాము. త్రిపద సమ్మేళన ప్రయోగాలలో మా పద్ధతి వెబ్ లేదా వ్యాససారాంశాలు వినియోగించే ఇతర పద్ధతులతో పోల్చదగిన ఖచ్చితత్వం అందిస్తుంది. చతుర్పద సమ్మేళనాలతో జరిపిన ముందస్తు పరిశీలనలు, మా విధానాలు మరింత పొడుగైన సమ్మేళనాల నిర్మాణ వర్గీకరణలో కూడా ఉపయోగపడే సామర్థ్యాన్ని సూచిస్తున్నాయి.

Keywords: సమ్మేళన నామవాచకాలు, వాక్య నిర్మాణ వర్గీకరణ, పరిభాషలు, జ్ఞానాధారిత పద్ధతులు

Proceedings of COLING 2012: Posters, pages 559–568,
COLING 2012, Mumbai, December 2012.

1 Introduction

Noun compounds are noun phrases that are comprised of tokens each of which is a noun. For example `cell count` or `colon cancer` are examples of NCs with two tokens. Although these examples are easy to interpret for a human reader, in general, the semantic content of a noun compound cannot be automatically extracted based on the constituent nouns. In the NCs `olive oil, baby oil`, and `fuel oil` the relationship between the second token 'oil' with the corresponding first tokens 'olive', 'baby', and 'fuel' is clearly different. However, in these cases, there is a way of deriving the meaning of the NCs using the constituent words. There are other non-compositional NCs, such as `baby boomer`, `snake oil`, and `olive branch`, where the meaning of the constituent tokens cannot be composed to arrive at the semantic interpretation of the corresponding NCs. In biomedical domains, we often see NCs with 3 or more tokens, where there is additional ambiguity with regards to the syntactic association among the constituent tokens that can lead to different semantic interpretations. Consider the NCs `cancer cell line` and `cancer cell apoptosis`. The first NC is the cell line (immortal cell sample) from a cancerous tumor and the latter is about the apoptosis (programmed cell death) of cancer cells. Thus, we see that, although the part-of-speech tags are exactly the same for both NCs, the syntax trees[1]

(cancer (cell line))   and   ((cancer cell) apoptosis)

are different. Since the semantic interpretation closely follows the syntactic interpretation (referred to as bracketing henceforth), it is an important task to correctly bracket NCs.

1.1 Motivation

Biomedical language processing poses several challenges including significant lexical variation, synonymy, polysemy, latent and implicit semantic content, and long sentences with long range compositional dependencies (Friedman and Johnson, 2006). NCs occur frequently in biomedical articles and clinical narratives, and are also found in labels of concepts in biomedical ontologies. Correctly bracketing NCs has applications in ontology alignment, semantic mappings, information extraction, question answering, and other informatics applications in biomedicine. In ontology alignment, identifying concept pairs from two different ontologies that are equivalent or involved in a specific relationship is an essential task. Concept labels together with interrelationships among concepts are used to achieve this goal. But these labels are often NCs and require appropriate interpretation to determine equivalence and identify relationships. NC analysis is also useful in generating semantic mappings where complex biomedical entities in relationships extracted from raw text need to be mapped to appropriate concepts in standard terminologies. Query expansion and modification using relevance feedback for recall oriented search tasks also benefit from NC analysis.

1.2 Related Work

Standard natural language processing tools do not exist for NC bracketing. Both chunkers and deep parsers — including the latest versions of Stanford parser (de Marneffe et al.,

[1] For NCs, these are always binary trees, also representable using binary bracketings. The number of possible ways of binary bracketing n elements is given by the famous n-th Catalan number $\frac{(2n)!}{(n+1)!n!}$.

2006) and Enju parser (Matsuzaki et al., 2007) — do not offer bracketing for NCs. Linguists and computer scientists have been studying NC bracketing mostly in non-biomedical domains in the recent past. Pustejovsky et al. (1993) used the frequencies of adjacent tokens in an NC to determine left or right bracketing for 3-token NCs. Lauer (1995) used the dependency model for NC bracketing based on frequencies of bi-grams in Grolier's encyclopedia achieving 80% accuracy on a dataset of 3-token NCs extracted from the encyclopedia. Recently, Keller and Lapata (2003) used the Web bi-gram counts and Girju et al. (2005) used decision trees for supervised NC bracketing to achieve similar results on Lauer's dataset.

Nakov and Hearst (2005) used new lexical surface features such as possessive markers, hyphenated or concatenated tokens, and capitalization and conducted several experiments using Web n-gram counts to achieve a 90% accuracy using a majority vote on the results of various techniques for Lauer's dataset. Their work is the first and the only attempt to perform bracketing of biomedical NCs. They also constructed a dataset of 430 three-token NCs from Medline[2] abstracts and achieved 95% accuracy using the majority vote of 23 different methods. Bergsma et al. (2010) used support vector machines with n-gram counts and binary lexical features to achieve an accurarcy of 88% with Nakov's dataset. Although these datasets contain NCs outside their original full context (e.g., the full sentences they occur in), the assumption made by all these efforts and our current effort is that effect of the context is not significant to identifying the bracketing option that corresponds to the most frequently used (or well accepted) interpretation. So the correct bracketing of an NC is assumed to be the one that reflects the compositional nature of its most frequent interpretation.

Currently, to the best of our knowledge, there are no attempts on bracketing 4-token NCs, although there were cumulative accuracy results for general noun phrases of arbitrary length by Pitler et al. (2010). Also, earlier 3-token NC bracketing methods are based on large corpora and have only been tested on non-biomedical datasets, with the exception of Nakov and Hearst (2005). Our approach is knowledge-based in that we only use the labels and definitions of UMLS[3] concepts to bracket biomedical NCs. Treating the label and concept definition set as a corpus, we bracketed NCs with techniques based on frequency and relatedness measures. We used the biomedical dataset used in the thesis by Nakov (2007) for 3-token NCs. We also tested our approach on separate 3- and 4-token NC datasets that we constructed by parsing biomedical abstracts (Section 3.1) since Nakov's set was mostly left bracketed. Our results indicate comparable performance to corpus based methods for the 3-token NCs and perform 40% better than random guessing for the 4-token NC dataset.

2 Knowledge-Based NC Bracketing Approach

We use the UMLS Metathesaurus (or just UMLS) as the knowledge base for the bracketing task. UMLS is an ongoing National Library of Medicine (NLM) effort that is an integration of 161 biomedical terminologies with about 2.6 million concepts and 8.6 unique concept names. A new version is released each year with updates from included source vocabularies and additional new terminologies. Besides maintaining the inter-concept relationships provided by the source vocbularies, UMLS also has concept mappings between different terminologies; synonyms for different concepts are also maintained. Thus, UMLS is an excellent source of terminological information in biomedicine. For this paper we particularly

[2] http://www.nlm.nih.gov/bsd/pmresources.html

[3] http://www.nlm.nih.gov/research/umls/

use the English unique concept names and the definitions (when provided) of concepts in UMLS.

Before we proceed, we enumerate the bracketing possibilities for 3- and 4-token NCs. As seen in example in Section 1, 3-token NCs usually have two options - left and right bracketing. However, 4-token NCs have five options. If $w_1\, w_2\, w_3\, w_4$ is an NC, where each w_i is a single-token noun, we have

$$((w_1 w_2) w_3) w_4, \quad (w_1 w_2)(w_3 w_4), \quad (w_1 (w_2 w_3)) w_4,$$
$$w_1((w_2 w_3) w_4), \quad \text{and} \quad w_1(w_2(w_3 w_4)),$$

as the five possible bracketing options.

2.1 Frequency Based Greedy Bracketing

There are nearly 6 million unique English concept names (ignoring case) in UMLS that encompass several important topics. We treat the set of these labels as a small corpus and count frequencies of token subsequences (based on word boundaries) of NCs to be bracketed. The first approach is to use the raw frequencies to choose the most frequent groupings. Let $f(x)$ be the frequency of the phrase x in the UMLS concept name corpus. For an NC with n tokens denoted by $w_1\, w_2 \ldots w_n$, the frequency function $f(w_i w_{i+1} \ldots w_j)$, $1 \leq i \leq j \leq n$, is the frequency of the phrase "$w_i\ w_{i+1} \ldots w_j$" in the corpus. For a 3-token NC $w_1\, w_2\, w_3$, if $f(w_1 w_2) > f(w_2 w_3)$, we choose left bracketing; otherwise, it is right bracketed. For 3-token NCs, we also employed the adjacency approach introduced by Pustejovsky et al. (1993) where instead of raw frequencies, simple proportions are used to determine left or right bracketing. Here, left bracketing is selected if $f(w_1 w_2)/f(w_2) > f(w_2 w_3)/f(w_3)$, otherwise right bracketing is chosen.

Algorithm 1 GREEDY-BRACKET-4NC (NC $w_1 w_2 w_3 w_4$)

```
Set $maxf = \max(f(w_1 w2), f(w_2 w_3), f(w_3 w_4))$
if $maxf = f(w_1 w_2)$ then
    if $\log_2(3).f(w_1 w_2 w_3) \geq f(w_3 w_4)$ then
        return $((w_1 w_2) w_3) w_4$
    else
        return $(w_1 w_2)(w_3 w_4)$
else if $maxf = f(w_2 w_3)$ then
    if $f(w_1 w_2 w_3) \geq f(w_2 w_3 w_4)$ then
        return $(w_1 (w_2 w_3)) w_4$
    else
        return $w_1((w_2 w_3) w_4)$
else
    if $\log_2(3).f(w_2 w_3 w_4) \geq f(w_1 w_2)$ then
        return $w_1(w_2(w_3 w_4))$
    else
        return $(w_1 w_2)(w_3 w_4)$
```

For 4-token NCs, we follow a greedy approach in choosing among the five possible options. Assuming $w_1\, w_2\, w_3\, w_4$ as the four-token NC, we use Algorithm 1 to choose the bracketing.

The intuition behind the algorithm is to use a bottom-up approach to bracket the most frequent adjacent token pair first, before bracketing longer subsequences. The pseudocode is mostly self explanatory, except that since these are raw frequencies, we use $\log_2(3)$ as a factor[4] to give more weight to the occurrence of three-token phrases when comparing them with two-token phrase frequencies.

In addition to using frequencies of NC tokens in the corpus of unique UMLS strings, we also experimented with frequency based and adjacency approaches using the set of all strings in the UMLS without ignoring duplicates arising out of identical concept labels from different terminologies. While considering unique strings gives more importance to the presence of a phrase in multiple unique UMLS labels, considering all strings gives more importance to the overall frequency with which a phrase appears in all labels, thus accounting for the association with multiple UMLS concepts.

2.2 Cohesion Measure Based Non-Greedy Bracketing

Raw frequency based approaches do not fully consider the relative frequencies of other tokens involved in an NC. For example, consider the NC `family health history`. Although the phrase 'family health' is more frequent than 'health history', we see that this NC is right bracketed as it is often interpreted as the health history of a family of an individual. Also, the greedy nature of the bracketing approach outlined in Algorithm 1 might not be ideal. For example, in the compound `liver membrane protein glycosylation`, the frequency of 'membrane protein' is higher than the frequencies of 'liver membrane' or 'protein glycosylation'. Using the greedy approach, (membrane protein) will be chosen as the first grouping. However, it turns out the correct bracketing has (liver membrane) as the first grouping with protein as its modifier. To counter this, we propose *bracketing cohesion* measures that provide a cohesion score based on the full structure of a bracketing choice. Once the cohesion measure is computed for all bracketing choices, the choice with the highest cohesion value is output as the correct bracketing.

Bracketing cohesion is a meta-measure based on other relatedness measures. Let $\mathscr{S}(t_1, t_2) \in [0, 1]$ be a measure that computes relatedness between any two given terms t_1 and t_2. Then, given a bracketing binary tree T, we define the corresponding bracketing cohesion measure

$$\mathbb{C}(T, \mathscr{S}) = \sum_{\text{non-leaf node } n \in T} \mathscr{S}(\text{left-child}(n), \text{right-child}(n)),$$

where left-node(n) and right-node(n) are the subsequences of NC tokens corresponding to the left and right children of node n. For example, let T be the bracketing tree shown in Figure 1 for the example used in this section. Then the cohesion measure value is $\mathscr{S}$(liver, membrane) + $\mathscr{S}$(liver membrane, protein) + $\mathscr{S}$(liver membrane protein, glycosylation).

Based on the cohesion values, the best bracketing is the one that corresponds to the bracketing tree T that maximizes[5] $\mathbb{C}(T, \mathscr{S})$. We note that this approach of using cohesion measures is generic and can be applied to NCs of any length. The intuition behind bracketing

[4] The general strategy is to use $\log_2$(# words in the term) as the weighting factor (Frantzi et al., 1998)

[5] When multiple trees have the same score, other ways of breaking the tie are needed; one can default to the most frequent bracketing tree in the observed data for that length. Also, the highest possible value for the cohesion measure for NCs of length n is $n-1$ since there are $n-1$ internal nodes and each $\mathscr{S}(t_1, t_2) \leq 1$.

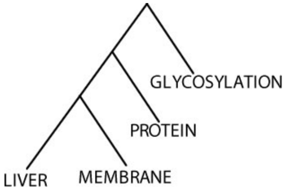

Figure 1: A bracketing tree for "Liver Membrane Protein Glycosylation"

cohesion is based on the observation that token subsequences in an NC that is compositional in nature are related to each other. Otherwise, they would not manifest in textual documents as NCs themselves and as parts of longer NCs. The intuition then is to model the relative suitability/validity of bracketing options for a given NC based on the strength of the relatedness between the subsequences that arise out of the tree structures corresponding to the bracketing options. For example, bracketing the NC in Figure 1 as (liver membrane) (protein glycosylation) would result in the cohesion value $\mathscr{S}$(liver, membrane) + $\mathscr{S}$(protein, glycosylation) + $\mathscr{S}$(liver membrane, protein glycosylation).

We experimented with three symmetric measures for $\mathscr{S}$. The first one is based on the Jaccard index – for two sets A and B, it is the ratio $\frac{|A \cap B|}{|A \cup B|}$, often used to measure resemblance of two sets of items. Translating this to the terms t_i and their frequencies $f(t_i)$, we have a measure

$$\mathscr{S}(t_1, t_2) = \frac{f(t_1 \wedge t_2)}{f(t_1) + f(t_2) - f(t_1 \wedge t_2)}.$$

Since this is a measure based on frequencies, we also used the UMLS label corpus with both unique strings and all strings which give us a total of two measures.

We also use a third measure that uses available concept definitions in the UMLS that are obtained from different source vocabularies and are more descriptive than the concept labels. Pedersen et al. (2007) derived second-order context vectors for UMLS concepts that capture the frequently co-occurring words in the definitions of concepts and certain concept neighbors (nodes reachable by one-hop), hence the name second-order, in the UMLS Metathesaurus relationship graph. They define a relatedness measure using the cosine of the normalized context vectors for any given UMLS concept pair. We call this measure UMLSRel[6] and use this as an option for $\mathscr{S}$ to compute cohesion in our experiments based on a local installation of the Perl modules made available by Pedersen et al. (2007). Other measures, such as mutual information can also be used for $\mathscr{S}$ when computing cohesion measures.

3 Experiments and Evaluation

We applied the methods elaborated in Section 2 to the biomedical 3-token NC dataset constructed by Nakov and Hearst (2005); Nakov (2007). This dataset has 430 biomedical NCs of which 84% are left bracketed. We separately constructed both 3-token and 4-token NC datasets that were bracketed by two biomedical researchers.

[6] If one of the terms does not correspond to a UMLS concept or if neither the term nor its neighbors have a definition, the relatedness value is treated as zero. This usually happens with longer terms with 3 or more tokens.

3.1 Construction of Datasets

We used Natural Language Tool Kit (NLTK (Bird et al., 2009)) to sample and parse approximately 175,000 biomedical research article abstracts from NLM's Pubmed web service. Using NLTK's chunker we sorted 3- and 4-token NCs based on their frequencies and manually selected 100 from each set according to the sorted order. For the 3-token NCs, the selection was done to maintain a rouch balance between possible left and right bracketed NCs while still going in the sorted order. Following the extraction, two biomedical researchers (not the authors) independently bracketed the datasets. The annotator agreement was 90% for our 3 NC dataset (NC) and it was 59% for the 4-token NC set (4NC). We note that expected agreement by chance is only 20% for 4-token NCs because of five possible choices, while it is 50% for the 3-token case. Since we started out with 100 NCs in each data set, we finally have 90 in the 3-token set and 59 in the 4-token set. Of the 90 three-token NCs, 42 are right bracketed; for the 59 NCs in the UK-4NC set, the bracketing choice $((w_1 w_2) w_3) w_4$ is the most frequent, with 32 instances, although, as explained in the next section, for 10 of these 32 cases annotators felt that the bracketing option $(w_1 w_2)(w_3 w_4)$ also applied. These gold standard bracketed NC files used for the experiments are provided here: `http://protocols.netlab.uky.edu/~rvkavu2/bracketing.html`.

3.2 Experiments and Discussion

For the 3-token NCs we used seven techniques to do the bracketing. The first four are the raw frequency based methods and the adjacency model based frequency proportion method outlined in Section 2.1, considering both the unique strings and all strings in the UMLS labels. These are denoted by `Freq, Adj, Freq_uniq, Adj_uniq` in Table 1. The next two methods are based on the bracketing cohesion method when the relatedness measure used in computing cohesion is based on the Jaccard index, again using all labels and only unique labels denoted by `Jaccard` and `Jaccard_uniq` respectively in the table. The final method uses the bracketing cohesion approach based on the context vector based UMLSRel (Pedersen et al., 2007) as discussed in Section 2.2. For the 4-token NCs, we used five techniques where the first two are based on the greedy frequency based bracketing approach outlined in Algorithm 1 using all UMLS strings and then using only unique strings separately. The next two methods pick the best bracketing option based on the cohesion measures of all possible bracketing options using the Jaccard index, again, using all strings and then only unique strings. Finally, the UMLSRel (Pedersen et al., 2007) measure is used for bracketing cohesion as outlined in Section 2.2. We also did a majority vote and defaulted to the most frequent option in the dataset to break ties. The results of these experiments are outlined in Tables 1 and 2.

From the results we see that frequency based approaches slightly outperformed other measures. The cohesion based methods slightly underperformed for the 3-token case compared to the frequency based measures. We attribute this to the nature of the measures chosen – both Jaccard index and UMLSRel are corpus based and moving beyond UMLS labels and definitions to corpus based approaches might be suitable. However, path based similarity measures based on the UMLS graph might be more suitable alternatives to be explored. We also computed majority vote based on our methods, which did not significantly improve the overall accuracy, although there were examples where some methods performed better than others. For the Nakov dataset, the majority vote with left bracketing as the tie-breaker improved the accuracy to 87% (up 3%). Nakov and Hearst (2005) use 23

Method	Nakov-3NC	3NC
Freq	84 %	89%
Freq_uniq	83%	85%
Adj	67%	78%
Adj_uniq	72%	77%
Jaccard	81%	75%
Jaccard_uniq	81%	74%
UMLSRel	79%	74%

Table 1: Accuracy for 3-token NCs

Method	4NC
Freq	63%
Fre_uniq	63%
Jaccard	48%
Jaccard_uniq	44%
UMLSRel	63%

Table 2: Accuracy for 4-token NCs

different methods in their majority vote for 3-token NCs to arrive at an accuracy of 95% on a significantly (84%) left bracketed dataset. It would be interesting future task to see how all those methods perform just by using the UMLS label set as the corpus. Coming to the 4-token NC dataset we constructed, our greedy frequency based approach is 41% more successful than random guessing that can lead to an expected 20% accuracy. In the dataset there were several contentious choices where researchers thought that there are two equally acceptable bracketing options. This happened in about 10 (out of 59) cases where the contention is between the choices $((w_1w_2)w_3)w_4$ and $(w_1w_2)(w_3w_4)$. An example of such an NC is `bone marrow cell proliferation`. Here annotators felt that both interpretations are appropriate. Accuracy improved from 63% to 70% when we allowed either choice for these contentious NCs.

4 Concluding Remarks

We pursued a knowledge-based approach to bracketing biomedical NCs with 3- and 4-tokens. In addition to employing frequency count based approaches, we also proposed the concept of bracketing cohesion that takes as input measures of term pair relatedness. We initially experimented with Jaccard's index and context vector based UMLSRel measures for computing bracketing cohesion. We plan to extend the bracketing cohesion using various other measures of relatedness including mutual information and also compute it over a bigger corpus. We would also like to explore other path based relatedness measures based on the UMLS graph structure. Although we don't have concrete results yet on entire dataset, using

$$\mathscr{S}(t_1, t_2) = \frac{1}{\text{shortest-path-length}(t_1, t_2)}$$

as the relatedness measure for cohesion based method (Section 2.2) produced good results for a smaller subset of the 3-token NCs. Another important frequency based measure that outputs term-hood scores to terms is the C-value method (Frantzi et al., 1998). We are currently in the process of computing C-values for different n-grams. The idea is to use the C-values instead of the frequencies in the greedy approach. We also plan to build and test our methods on a larger 4-token NC dataset and perform a more thorough analysis on inter-annotator agreement and confidence intervals for accuracies on unseen datasets.

Acknowledgements

Many thanks to Jacob Painter and Supriya Prabhala for helping us create the gold standard dataset. The project described was supported by the National Center for Research

Resources, UL1RR033173, and the National Center for Advancing Translational Sciences, UL1TR000117. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

References

Bergsma, S., Pitler, E., and Lin, D. (2010). Creating robust supervised classifiers via web-scale n-gram data. In *ACL*, pages 865–874.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

de Marneffe, M., MacCartney, B., and Manning, C. (2006). Generating Typed Dependency Parses from Phrase Structure Parses. In *Proceedings of LREC 2006*.

Frantzi, K. T., Ananiadou, S., and Tsujii, J.-i. (1998). The c-value/nc-value method of automatic recognition for multi-word terms. In *Second European Conf. on Research and Advanced Tech. for Digital Libraries*, ECDL '98, pages 585–604.

Friedman, C. and Johnson, S. B. (2006). Natural language and text processing in biomedicine. In Shortliffe, E. H. and Cimino, J. J., editors, *Biomedical Informatics*, Health Informatics, pages 312–343. Springer New York.

Girju, R., Moldovan, D., Tatu, M., and Antohe, D. (2005). On the semantics of noun compounds. *Comput. Speech Lang.*, 19:479–496.

Keller, F. and Lapata, M. (2003). Using the web to obtain frequencies for unseen bigrams. *Comput. Linguist.*, 29:459–484.

Lauer, M. (1995). *Designing Statistical Language Learners: Experiments on Noun Compounds*. PhD thesis, Macquarie University, Australia.

Matsuzaki, T., Miyao, Y., and Tsujii, J. (2007). Efficient HPSG parsing with supertagging and CFG-filtering. In *Proceedings of the 20th international joint conference on Artificial intelligence*, IJCAI'07, pages 1671–1676.

Nakov, P. and Hearst, M. (2005). Search engine statistics beyond the n-gram: Application to noun compound bracketing. In *Proceedings of CoNLL-05*, pages 17–24.

Nakov, P. I. (2007). *Using the Web as an Implicit Training Set: Application to Noun Compound Syntax and Semantics*. PhD thesis, Univ. of California, Berkeley.

Pedersen, T., Pakhomov, S. V. S., Patwardhan, S., and Chute, C. G. (2007). Measures of semantic similarity and relatedness in the biomedical domain. *J. of Biomedical Informatics*, 40:288–299.

Pitler, E., Bergsma, S., Lin, D., and Church, K. W. (2010). Using web-scale n-grams to improve base np parsing performance. In *COLING*, pages 886–894.

Pustejovsky, J., Anick, P., and Bergler, S. (1993). Lexical semantic techniques for corpus analysis. *Comput. Linguist.*, 19:331–358.