StringNet as a Computational Resource for Discovering and Investigating Linguistic Constructions

David Wible **Nai-Lung Tsao**
National Central University
No.300, Jhongda Rd.
Jhongli City, Taoyuan County 32001, Taiwan
wible@stringnet.org beaktsao@stringnet.org

Abstract

We describe and motivate the design of a lexico-grammatical knowledgebase called StringNet and illustrate its significance for research into constructional phenomena in English. StringNet consists of a massive archive of what we call hybrid n-grams. Unlike traditional n-grams, hybrid n-grams can consist of any co-occurring combination of POS tags, lexemes, and specific word forms. Further, we detect and represent superordinate and subordinate relations among hybrid n-grams by cross-indexing, allowing the navigation of StringNet through these hierarchies, from specific fixed expressions ("It's the thought that counts") up to their hosting proto-constructions (e.g. the It Cleft construction: "it's the [noun] that [verb]"). StringNet supports discovery of grammatical dependencies (e.g., subject-verb agreement) in non-canonical configurations as well as lexical dependencies (e.g., adjective/noun collocations specific to families of constructions).

1 Introduction

Constructions have posed persistent challenges to the field of computational linguistics (Baldwin et al 2004; Sag et al 2002; Zhang et al 2006). Challenges to both statistical and symbolic approaches arise, for example, from the meager degree of productivity and non-canonical structures of many constructions and, as a loosely defined family of linguistic phenomena, their varied mix of regularity and idiomicity (Fillmore, Kay, and O'Connor 1988). It has been argued for decades that constructions are central rather than peripheral to any adequate account of linguistic knowledge and that they pose substantial challenges to mainstream accounts of language (Bolinger, 1977, 1985; Fillmore, Kay, and O'Connor, 1988; Goldberg, 1995; inter alia). But the recent attention they have been receiving in computational research is perhaps due more to their status as troublemakers (or a "pain in the neck", Sag et al 2002). Baldwin et al (2004) found, for example, that 39% of parse failures on clean data (BNC) occurred on constructions. (See Zhang et al (2006) for other such findings.) Thus, it is becoming urgent to "deal with" constructions for the sake of NLP. In this paper, however, we would like to shift perspective a bit to explore instead the application of computational resources for the sake of constructions. Our longer term aim is to broaden and deepen research on constructions in order to support the learning and teaching of constructions in second language education. Two basic challenges we address are: (1) the varied mix of regularity and idiomicity to be found within the wide range of constructions in a language (Fillmore, Kay, and O'Connor, 1988; Jackendoff, 2008 inter alia), and (2) the inheritance-like hierarchical relations holding between and among different constructions as instances of more general constructions or proto-constructions subsuming other constructions as sub-cases (Goldberg 1995 inter alia). To address these, we introduce a lexico-grammatical knowledgebase called StringNet and describe some ways that it can support the investigation of constructions.

Within the broad range of definitions for constructions, one widely shared premise is that the traditional division between lexical knowledge on the one hand and grammatical rules on the other is an artificial one. There are huge tracts of linguistic territory lying between the lexical and the grammatical which usage-attuned linguists have seen as not simply a residue of undiscovered deeper general principles but as the actual lay of the linguistic land (Bolinger 1977). We have taken this lexico-grammatical territory as a core target of the work we report here. StringNet has been designed to

Proceedings of the NAACL HLT Workshop on Extracting and Using Constructions in Computational Linguistics, pages 25–31,
Los Angeles, California, June 2010.

provide traction on some of this intermediate terrain.

The paper is organized as follows. Section 2 describes and motivates the basic approach we have taken in designing StringNet. Section 3 describes the design of StringNet itself. In Section 4, we illustrate the significance of StringNet for construction research with some extended examples. Section 5 is the conclusion.

2 Background and Approach

The specific approach we take to designing StringNet is motivated by the varied mixture of idiomicity and regularity exhibited by constructions mentioned above and the problems this poses both for symbolic and statistical approaches in computational linguistics. To frame the properties of constructions that we hope StringNet can help address, we make use of Fillmore, Kay, and O'Connor's distinction between substantive and formal idioms (1988), the latter of which they categorize eventually under "grammatical constructions" (p. 506). Substantive (or "lexically filled") idioms are those fixed at the lexical level, that is, lexical strings relatively frozen except perhaps for inflectional variation. Among examples they site are *pull a fast one*, *all of a sudden*, *kick the bucket*. Others, extracted by StringNet, would include *as a matter of fact*, *at a moment's notice*, *just to be on the safe side*, and a massive inventory of other fixed strings. In contrast to substantive idioms, formal (or "lexically open") idioms "…are syntactic patterns dedicated to semantic and pragmatic purposes not knowable from their form alone" (p. 505) These would include such expressions detected with StringNet as "bring [pnp[1]] to [dps] senses," "stop [pnp] in [dps] tracks," "It is safe to [verb] that" (e.g., *It is safe to assume/say/predict that*)," "There is a tendency for [noun] to [verb]," "[verb][dps] socks off" (e.g., *knock your socks off*). As mentioned above, on Fillmore et al's analysis, it is the latter type, the formal idioms, which are eventually "absorbed into the category of grammatical constructions" (p. 506). Crucially for us, however, they point out the potential significance of substantive (lexically filled) idioms for construction research. A substantive or frozen idiom may be a sub-case of a formal or lexically open idiom. Our example of this is the lexically filled idiom "It's the thought that counts" with its idiosyncratic interpretation that must be learned as a listeme; it presupposes something substandard about a gift or an effort as well as forgiveness of this in light of the good intentions of the giver. Yet much of its meaning derives from its status as an instance of the more general "It cleft" construction; the focus slot hosts one member of a contrasting pair or set, and that member is assumed to be new information, etc.).

Considering the challenges of extracting and representing these two sorts of expressions, substantive idioms have been the far more tractable of the two. Specifically, substantive, lexically filled idioms are readily susceptible to detection and representation by traditional n-grams. It is formal (lexically open) idioms, however, which have been identified more closely with constructions, yet they have proven much more resistant to extraction by computational means; for example, approaches using n-grams have so far shown little progress in handling this category of expression. And parsers famously have difficulties with their non-canonical structures (Baldwin et al 2004; Zhang et al 2006; inter alia).

The design of StringNet is aimed at addressing three long-recognized qualities of constructions: (1) the non-canonical structures of many of them; (2) their syntagmatic mixing of fixed and substitutable slots, making them resistant to representation by traditional n-grams; and (3) the hierarchical relations holding among them, as, for example, "it's the thought that counts" instantiates the general It Cleft construction while each arguably warrants independent status as a construction.

3 Design and Construction of StringNet

3.1 Overview

In this section we describe the design of StringNet. In light of the well-documented problems that constructions pose for parsers, we eschew parsing at this stage to see what we can achieve without it first.[2] StringNet is a corpus-derived knowledge-

[1] The glosses for the POS tags appearing in the paper, taken from CLAWS 5 tagset is are follows: pnp = pronoun, dps = possessive determiner, nn1 = singular noun, nn2 = plural noun, vvz = present 3rd person singular verb; vm0 = modal verb.

[2] StringNet will provide some natural spaces where shallow parsing could play a well-motivated role, but we leave that for future work.

base, automatically extracted from the British National Corpus (BNC). The structure of StringNet can be described in two parts: (1) a special type of n-grams that we refer to as hybrid n-grams, constituting the core content of StringNet and (2) the inter-relations among these hybrid n-grams, represented by cross-indexing. We describe and motivate these two aspects in turn.

3.2 Hybrid n-grams

Unlike traditional n-grams, hybrid n-grams can consist of co-occurring grams from different levels of analysis, more specifically, a combination of lexemes, word forms, and parts of speech (POSs) potentially side by side within the same string. For example, "from my point of view" is a traditional n-gram attested in BNC, where the grams are all lexical. However, our hybrid n-gram extraction, in addition, detects the substitutability of the second slot in this string and indicates this substitutability by a POS in that position: "from [dps] point of view". By including POS categories, hybrid n-grams can encode the paradigmatic dimension in addition to the syntagmatic one represented by traditional n-grams.

The hybrid n-grams that constitute StringNet's content are derived from BNC. Specifically, we include any contiguous combination of gram types ranging from bi-grams to 8-grams. Two criteria must be met for each hybrid n-gram. (1) It must include at least one lexical gram in the string (that is, either a lexeme or a specific word form). This means that all of the hybrid n-grams are "lexically anchored" to some extent. And (2) it must be attested in BNC at a minimum frequency of five tokens.

There are four categories of grams that can occur in the hybrid n-grams of StringNet. From specific to general, these categories are: (1) word form (thus, *ran*, *run*, and *running* are three distinct word forms); (2) lexeme (**run**, including all its different inflectional forms: *run*, *ran*, *running*); these are indicated in bold to distinguish them from word forms; (3) detailed POS category, taken from the large CLAWS set of 46 tags ([nn1] for singular noun); these are marked off in brackets; (4) rough POS category, taken from abbreviated tagset of 12 POS tags ([noun], including plural and singular nouns); indicated with brackets as well to avoid flooding users with too many distinctions in the representations. Further, each hybrid n-gram is indexed to all tokens instantiating it in BNC. Thus, every token of "saw the light" occurring in BNC is indexed to all hybrid n-grams that it satisfies, for example, indexed to "[verb] the light", "**see** [det] light", "[verb] [det] light", "saw the [noun]", and so on. As mentioned above, only hybrid n-grams attested by at least five tokens occurring in BNC are kept in StringNet.

3.3 Structure of StringNet: Cross-indexing of Hybrid n-grams

Since the inventory of gram types consists of four categories and these can stand in subordinate and superordinate relation to each other, it becomes possible to find relations of inclusion or subsumption between hybrid n-grams. For the sake of simplicity in the user interface, we label these as parent/child relations.

Take the tri-gram "paying attention to" as an example. As a string of word forms, this hybrid n-gram can be considered a child of the hybrid n-grams: **pay** attention to (where **pay** indicates the lexeme and includes forms pay, paid, paying). Non-monotonically, then, "paying attention to" can (and does) have more than one parent, for example: **pay** [noun] to; **pay** attention [prep]; among several others. StringNet exhaustively cross-indexes all of these thus-related hybrid n-grams. (Note that hybrid n-grams can have more complicated relations with each other, but these are not indexed in the current StringNet.) As a massive inventory of hybrid n-grams and the cross-indexing among them, StringNet is very large. For comparison, the size of our POS-tagged BNC is 4.4 GB. StringNet, which we extracted from BNC, is over a terabyte (over 1,000 GB), about 250 times the size of BNC.

The hybrid n-grams making up StringNet were extracted from BNC on the simple criterion of frequency (minimum frequency of 5 tokens in BNC), making no use of statistical techniques such as word association measures in the extraction process. However, to support queries of StringNet we must have some criteria for ranking the hybrid n-grams returned in a query result. For this, we use MI as our default hybrid n-gram association measurement. The MI equation is as follows:

$$MI = \log\left(\frac{P(X)}{P(x_1)P(x_2)...P(x_n)}\right)$$

,where $X = x_1 x_2 ... x_n$

This equation is well-known as an association measure for collocations consisting of word pairs. However it is not appropriate directly used in measuring hybrid n-grams or n-grams in LexChecker because it cannot compare n-grams of different length, i.e with different values of n. It would typically be biased toward longer n-grams. Therefore we use a version which normalizes, as follows:

$$Normalized\ MI(h_n, q) = \frac{MI(h_n)}{\max MI_n(q)}$$

,where h_n is the target hybrid n-gram, q is user query, $MI(\ )$ is the traditional MI equation mentioned above and maxMI_n is the maximum MI score achieved among all of the n-grams of any given length n and retrieved for query q.

For example, a hybrid tri-gram T="pay attention to" and a hybrid 4-gram Q="pay attention to the" will be shown in the results of the query q="attention". Assume $MI(T)$=5, $MI(Q)$=7, maxMI_3 ("attention") =15 and maxMI_4 ("attention") = 20. Then the *Normalized* $MI(T,q)$ = 5/15 = 0.334 and *Normalized* $MI(Q,q)$ = 7/20 = 0.35. So we can rank Q higher than T. $MI(h_n)$ will never be greater than max$MI_n(q)$ because by stipulation, max$MI_n(q)$ represents the highest MI score of all n-grams at a given value of n and a query q. So Normalized MI will always fall between 0 and 1. This creates a common specified range within which MI scores for hybrid n-grams of different lengths can be ranked. It is important to note that this ranking measure is not incorporated into StringNet itself (e.g., as a criterion for hybrid n-grams to be included in StringNet). Rather it is a post hoc means of ranking search results. StringNet is compatible with other methods of ranking and contains all statistical information needed to run such alternative measures.

3.4 Pruning

As we mention above, hybrid n-grams in StringNet consist of all possible combinations of word form, lexeme and two types of POS in strings from 2 to 8 grams in length. Thus for every single traditional n-gram consisting of a string of word forms, there are numerous hybrid n-grams that also describe that same string. For a traditional 8-gram, for example, we create $4^7 \times 2$=32768 different hybrid n-grams (taking into account our criterion that at least one token has to be a word form or lexeme). Such a large amount of information will cause low performance of the StringNet applications. In order to decrease the search space while still keeping most of the useful information, we introduce pruning. Specifically, pruning is intended to eliminate redundant hybrid n-grams from searches or applications of StringNet. There are two types of pruning we use in StringNet currently: Vertical pruning and Horizontal pruning.

Vertical pruning:

Vertical pruning considers pairs of hybrid n-grams that are identical in length and differ in the identity of some gram in the sequence. Consider the following such pair.

a. hybrid n-gram 1: my point [prep] view
b. hybrid n-gram 2: my point of view

These 4-grams are identical except for the third gram; moreover, the counterpart grams occupying that third slot ("of" and [prep]) stand in an inclusion relation, "of" being a member of the POS category [prep]. Recalling our cross-indexing, this parenthood relation between such hybrid n-grams can be readily detected. Pruning of the parent occurs in cases where a threshold proportion of the instances attested in BNC of that parent are also instances of the child. Consider (a) and (b) above. Here the parent (a) "my point [prep] view" would be pruned since all cases of [prep] in this pattern in BNC are indeed cases of the preposition "of". Consider now (c), another parent hybrid n-gram of (b) that, in contrast, would not be pruned.

c. hybrid n-gram 3: [dps] point of view

This parent is retained because "my" accounts for fewer than 80% of the instances of the [dps] in this pattern. The retention of "[dps] point of view" indicates that more than one possessive pronoun is attested in the [dps] slot of this string in a threshold proportion of its cases and thus the slot shows sub-

stitutability. In a word, vertical pruning eliminates hybrid n-grams containing POS grams which do not represent attested substitutability. Currently, for our StringNet search interface (LexChecker) we prune parents with children that represent over 80% of the BNC tokens also described by that parent.

Horizontal pruning:
The main idea of Horizontal pruning is the same as Vertical pruning. The only difference is the axis of comparison: For horizontal pruning, two hybrid n-grams for comparison differ only by value of n (ie., by length). For example, comparing the hybrid n-gram "[dps] point of" and "[dps] point of view," the shorter one is parent and is pruned if a threshold proportion of its instantiations in BNC are also instances of the longer child "[dps] point of view." In horizontal pruning, the shorter of the two compared hybrid n-grams is the potentially redundant one and thus the candidate for pruning. As with our MI measure, both vertical and horizontal pruning rate are set post hoc, applied by post-processing, and so are adjustable.

4 Illustrating with Examples

Although StringNet can support a wide range of applications (such as error detection and correction (Tsao and Wible 2009); document similarity measurement, etc.), for ease of exposition in what follows, we take a search query as our access point to illustrate StringNet content. Taking *eye* as our query term, StringNet yields a ranked list of 3,765 hybrid n-grams containing either this lexeme or one of its inflected forms. The following are samples from the top 50 (i.e., the first page of results):

visible [prep] the naked eye
turning a blind eye to
out of the corner of [dps] eye
[dps] eyes filled with tears
keeping an eye on the [noun]
[adv] see eye to eye
look [pers prn] straight in the eye
cast a [adj] eye [prep] (e.g., *cast a critical eye over*, *cast a cold eye on*)

Each hybrid n-gram listed in a search result is accompanied by links to examples and parent and child icons that link to its parent and children hybrid n-grams. (See Fig 1 and 2.) Consider one of the hybrid n-grams listed in the results for *eye*: "keep a close eye on." Recalling Fillmore et al's distinction between substantive and formal idioms, in the case of "keep a close eye on" we are at the level of the formal (lexically filled) idiom. Note that since it is a string of lexical items, as are all substantive idioms by definition, this sort can just as easily be extracted and represented using traditional flat n-grams. StringNet's hybrid n-grams and their cross-indexing, however, allow us to see whether this is a one-off lexically filled idiom or an instance of a lexically open formal idiom (i.e., of a construction). Without hybrid n-grams, the next step up in abstraction to determine this would be pure POS n-grams (strings of POS categories only) used in the literature (Feldman et al 2009; Florian et al 2003; Gamon et al 2009). In the case of "keep a close eye on" the corresponding POS n-gram would be "[verb][det][adj][noun][prep]." This, however, could describe strings as far afield as "buy a new car with" or "sequester the entire jury until." Our hybrid n-grams are intended to address this Goldilocks problem where constructional phenomena fall between these two sorts of traditional n-gram representations evading detection by both.

No	Hybrid ngram	Examples	Parents	Children
1.	**keep** a [aj0] eye on the [nn1]			
2.	**keep** a [aj0] eye on			
3.	[cjc] **keep** a [aj0] eye on			
4.	keep a [aj0] eye on			
5.	**keep** a [aj0] eye on the			

Figure 1: StringNet search interface: "keep a [adj] eye on"

No	Hybrid ngram	Examples	Parents	Children
1.	keeping a [aj0] eye on			
2.	**keep** a close eye on			
3.	**keep** a watchful eye on			
4.	kept a [aj0] eye on			
5.	keep a [aj0] eye on			

Figure 2: Children of "keep a [adj] eye on"

Navigating from "keep a close eye on" upward through the pruned StringNet network using the parent and child links, we find the parent "keep a [adj] eye on" instantiated by attested examples "keep a close/watchful/wary/keen eye on." Another parent of "keep a close eye on" is "keep a close [noun] on".

Tellingly there are only two nouns attested more than once in the noun slot in this frame: "keep a close eye/watch on." Both of these parents in turn share the common parent "keep a [adj][nn1] on." This parent is attested by 268 tokens in BNC. Among these, there are 80 distinct [adj][nn1] pairings filling those two POS slots in this hybrid n-gram (e.g., *close eye*, *firm grip*, *tight rein*, *close watch*, etc.). StringNet allows the extraction of this set of 80 [adj][nn1] pairs and indexes this set to this specific hybrid n-gram. This enables a range of investigations. One direction from here is to explore this particular set of 80 [adj][noun] pairs. For example, we could take this set of pairs as a potential identifying feature set of this construction and search StringNet for other hybrid n-grams with the substring [adj][noun] to identify those that show a large overlap with the 80 pairs from "keep a [adj][noun] on." This would constitute an approach to detecting similar constructions or family resemblances between and among constructions. Another direction is to see whether "keep" is an anchoring lexical element of this construction or substitutable much like the [adj] and [noun] slots. This could be investigated in a number of ways in StringNet. For example, by comparing "keep a [adj][noun] on" with minimally distinct hybrid n-grams with verbs other than "keep," conditional or relative probability measures could indicate whether that set of 80 [adj][noun] pairs from "keep a [adj][noun] on" is conditioned by "keep" or independent of the particular verb in this string.

It's the thought that counts:

For this example, we query StringNet for "count" and get 436 distinct, unpruned hybrid n-grams for the verb. The eight listed below include the top-ranked 5 with 3 others sampled from the top 12, rank order retained:

stand up and be counted
count the number of [nn2]
count [dps] blessings
it **be** the [noun] that **count**
[vm0][adv] be counted as
[pnp] [vm0] not count on
what counts as [nn1]
count [pronoun reflx] lucky

Ranked 4th among these is "it **be** the [noun] that **count**," attested with 21 tokens in BNC. In 9 of these tokens, the [noun] is *thought*, so of course, navigating down we find "it's the thought that counts" as a descendant hybrid n-gram. Numerous aspects suggest themselves. First is the relation between lexically filled substantive idioms and more abstract formal idioms that host them. Starting with the lexically filled "it's the thought that counts" and navigating upward we note that *count* remains specified but can host a range of nouns in the focus position, as indicated by our 4th ranked "it **be** the [noun] that **count.**" The nouns attested in this slot are: *hunt, perception, topic, message, future, critic, change, books, feelings, character, voter, sport*. Upward from here to a proto ancestor, we reach "it be the [noun] that [verb]," a bare-bones frame of the It Cleft construction and host to the generations of instantiations below it.

Dependency Discovery

In addition to relations among constructions that StringNet encodes, it also yields up internal dependencies between co-occurring grams within a construction. A grand-daughter of the proto "It Cleft" string is telling in this respect: "it **be** the [nn1] that [vvz]". In other words, StringNet here indicates morphological agreement in the "It Cleft" construction. Statistical work on the tokens of these hybrid n-grams can detect such dependencies automatically. Crucially, StringNet provides traction on the grammatical features of quirky aspects of constructions, that terrain between regularity and idiomicity that poses such persistent problems for NLP.

5 Conclusion

StringNet has been created as a resource for investigating constructions and a range of multiple word expressions and for supporting NLP applications that traffic in constructions. While StringNet has been extracted from BNC, we hope that in turn StringNet can provide a richer setting for investigating a range of linguistic phenomena. For example, while computational techniques for extracting collocations have been run on traditional corpora,

deeper and more finely nuanced collocation knowledge can be discovered when the larger context of a framing construction is taken into account. Thus not just extracting [adj][noun] collocations, but ones particular to a framing construction or family of constructions. StringNet also renders up grammatical dependencies otherwise hard to detect since they are within the non-canonical structures of constructions. It is hoped that further cross-indexings of StringNet in the future can support increasingly nuanced research on constructions.

Acknowledgments

The work described in this paper was partially supported by the grants from the National Science Council, Taiwan (Project Nos. 96-2524-S-008-003- and 98-2511-S-008-002-MY2).

References

Timothy Baldwin, Emily M. Bender, Dan Flickinger, Ara Kim and Stephan Oepen. 2004. Road-testing the English Resource Grammar over the British National Corpus. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, pp. 2047-2050.

Dwight Bolinger. 1977. Idioms Have Relations. *Forum Linguisticum* 2:157-69.

Dwight Bolinger. 1985. Defining the Indefinable. In Robert Ilson (ed.) *Dictionaries, Lexicography, and Language Learning, ELT Documents 120*. Oxford: Pergamon Press, pp. 69-73.

Gosse Bouma and Begona Villada.2002. Corpus-based acquisition of collocational prepositional phrases. In *Proceedings of Computational Linguistics in the Netherlands (CLIN)* 2001, University of Twente.

Sergey Feldman, Marius Marin, Julie Medero and Mari Ostendorf. 2009. Classifying Factored Genres with Part-of-Speech Histograms. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Boulder, Colorado, pp. 173-176.

Charles J. Fillmore, Paul Kay, and Mary Katherine O'Connor. 1988. Regularity and Idiomaticity in Grammatical Constructions: the Case of *Let Alone*. *Language* 64: 501–538.

Radu Florian, Abe Ittycheriah, Hongyan Jing and Tong Zhang. 2003. Named Entity Recognition through Classifier Combination. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003.*

Michael Gamon, Claudia Leacock, Chris Brockett, William B. Dolan, Jianfeng Gao, Dmitriy Belenko, and Alexandre Klementiev. 2009. Using Statistical Techniques and Web Search to Correct ESL Errors. *CALICO Journal,* 26(3), pp 491-511.

Adele Goldberg, 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago: University of Chicago Press.

Ray Jackendoff 1997. The Boundaries of the Lexicon. in M. Everaert, E.-J. van der Linden, A. Schenk, and R. Schreuder, eds., *Idioms: Structural and Psychological Perspectives*, 133-165. Hillsdale, NJ: Erlbaum.

Ivan Sag, Timothy Baldwin, Francis Bond, Ann Copestake, Dan Flickinger. 2002. Multiword expressions: A pain in the neck for NLP. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002)*, Mexico City, Mexico, pp. 1-15.

Nai-Lung Tsao and David Wible. 2009. A Method for Unsupervised Lexical Error Detection and Correction. *The NAACL Workshop on Innovative Use of NLP for Building Educational Applications*, Boulder, Colorado, pp. 51-54.

Yi Zhang, Valia Kordoni, Aline Villavicencio, Marco Idiart. 2006. Automated Multiword Expression Prediction for Grammar Engineering. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties. COLING-ACL 2006*. Sydney. Australia.