Bootstrapping Open-Source English-Bulgarian Computational Dictionary

Krasimir Angelov

University of Gothenburg
Göteborg, Sweden
krasimir@chalmers.se

Abstract

We present an open-source English-Bulgarian dictionary which is a unification and consolidation of existing and freely available resources for the two languages. The new resource can be used as either a pair of two monolingual morphological lexicons, or as a bidirectional translation dictionary between the languages. The structure of the resource is compatible with the existing synchronous English-Bulgarian grammar in Grammatical Framework (GF). This makes it possible to immediately plug it in as a component in a grammar-based translation system that is currently under development in the same framework. This also meant that we had to enrich the dictionary with additional syntactic and semantic information that was missing in the original resources.

Keywords: English, Bulgarian, Translation, Morphology, Open-Source

1. Introduction

A good translation dictionary is one of the basic linguistic resources that is often needed but is not always easy to get. It can be used for improving machine translation or just as a language learning resource for humans. Although translation dictionaries exist for many language pairs they are often proprietary which makes them inaccessible for many purposes.

We present an open-source English-Bulgarian dictionary with LGPL license. The dictionary can be used both as monolingual morphological lexicon and as translation dictionary. We bootstrapped the dictionary from existing open-source resources, and we enriched it with additional linguistic information.

The dictionary is created in the Grammatical Framework GF (Ranta, 2011) and is compatible with the already existing Resource Grammar Library (Ranta, 2009) in GF. The library contains wide coverage grammars for currently about thirty languages which are linked together by using a common abstract syntax. Parsing a sentence with one of the grammars results in an abstract tree which could be linearized in any of the other languages. By doing this we get a baseline translation system. On top of the library, it is possible to build application specific grammars which provide better translations in specific domains. A central resource in this translation pipeline is the translation dictionary which is currently under development for Bulgarian, Chinese, English, Finnish, French, German, Hindi, Italian, Urdu, Spanish and Swedish. In this paper, we focus only on the English–Bulgarian part of the dictionary.

Since the framework can export the dictionary in external formats, it is possible to reuse the resource in other frameworks and projects.

We also build an Android application which is a front-end to the dictionary and to the GF based translation.

2. English Lexicon

We started by building a monolingual English lexicon. There are plenty of freely available resources for English so this is not particularly difficult. We used two main sources: the computer usable version of Oxford Advanced Learners Dictionary (OALD) (Mitton, 1986) and the Princeton WordNet (Fellbaum, 1998).

The first is a dictionary of English in computer-usable form, extracted from the third edition of OALD (Hornby, 1974). It contains about 40 000 lemmas together with a part of speech tag and a full-form inflection table. The glosses for the lemmas are not included and in that sense this is only partial representation of OALD but in return the resource is ready for use in linguistic applications and it is freely distributed unlike the original OALD.

From WordNet we extracted all words which were not already in OALD. This gave us very good coverage for verbs, adjectives, adverbs and nouns. Unfortunately WordNet does not provide any morphological information. We compensated for this by using the automatic inflection in GF.

In other words, whenever we know the inflection table from OALD, we generate an entry like:

```
book_N = mkN "book" "books"
```

Here `book_N` is a unique identifier which we generate by combining the lemma of the word with its part of speech tag. The list of all tags is shown in Table 1. The rest says that this is a noun with the corresponding inflection forms. `mkN` itself is an overloaded function defined in the English grammar. It takes different number of arguments and depending on that it does slightly different things. In this case it is applied to two strings which are interpreted as the singular and plural forms for the noun. Based on that the function automatically infers the genitive forms which are always predictable in English. The final result is that the lemma identifier is mapped to a table which is computed by `mkN` and contains all possible forms of the word.

When the word comes from WordNet, then we know only the base form for the word, and we generate an entry which contains only that form:

```
pachinko_N = mkN "pachinko"
```

Since now we apply mkN to only one argument, this tells the library that all other forms need to be inferred automatically. Détrez and Ranta (2012) have shown that by using

N	noun
Pron	pronoun
A	adjective
Adv	adverb
AdA	adjective-modifying adverb
AdN	numeral-modifying adverb
AdV	adverb directly attached to verb
CAdv	comparative adverb
V	intransitive verb
V2	transitive verb
V3	ditransitive verb
V0	impersonal verb
VV	verb with verb complement
VS	verb with sentence complement
VQ	verb with question complement
V2V	verb with object and verb complement
V2S	verb with object and sentence complement
V2Q	verb with object and question complement
Det	determiner
Quant	quantifier
Predet	predeterminer
IDet	interrogative determiner
IP	interrogative pronoun
IAdv	interrogative adverb
Conj	conjunction
Prep	preposition

Table 1: Part of speech tags

the morphological functions in the English grammar, it is possible to infer the right inflection for 95% of the nouns and 84% of the verbs by stating only the lemma. In addition GF provides a list of irregular verbs which we have used to capture eventual irregularities. Since the automatic method is not perfect it is possible that some wrong forms have slipped through but those will be fixed when they are spot. We believe that most of the irregular words have been included in OALD.

A small number of closed-class words such as prepositions, conjunctions and a number of multi-word expressions, we extracted from the Penn Treebank (Marcus et al., 1993) and from Wikipedia. The extraction from the Penn Treebank came as a side effect from an ongoing work to match the Penn Treebank with the English grammar in GF. For instance single word prepositions and conjunctions can be extracted by just searching through the corpus for words with the corresponding part of speech tag. In addition, we found that a number of parsing failures were caused by non compositional multi-word units such as 'because of', 'as well as', etc. These word sequences cannot be parsed by the grammar unless if they are explicitly listed in the dictionary. We detected some of those from the corpus. In addition there is a Wikipedia article[1] listing common multi-word prepositions in English.

The GF Resource Grammar Library has more detailed sub-categorization for verbs and adverbs than we could find in either OALD or WordNet (see Table 1). This means that we had to enrich the original resources.

[1] http://en.wikipedia.org/wiki/List_of_English_prepositions

For the verbs, we extracted valency sub-categorization by analysing the syntactic trees in the Penn Treebank where each verb is used. Independently Dannells and Gruzitis (2014) initiated a project for extracting semantic grammar for GF from FrameNet (Ruppenhofer et al., 2005). There, a central problem is again the representation of verb valency in GF. Certainly there is an overlap and we expect that the valency frames in FrameNet are of higher quality, but it is still a remaining task to use their extraction algorithm in our translation lexicon.

For adverbs the grammar library provides a small lexicon of structural words where several adverbs are listed with their corresponding sub-categorization. We incorporated this lexicon into our bigger lexicon, and we added more sub-categorizations by comparing the annotated trees in Penn Treebank with the GF grammar.

The lexicon has also undergone a continuous cleanup for inconsistencies that we spot in the primary sources. For example, the English definite article was listed as an adverb in OALD. This is simply abuse of the available tags. It comes from the phrase 'the more ... the more ...' which in later editions is marked as an idiom while in the third edition is marked as an adverb. All numeral words are listed as both adjectives and nouns in both OALD and WordNet. For numerals the English GF grammar has a special category with corresponding syntactic rules. We just removed those words from the lexicon and instead we rely on the grammar. Whenever we spot entries which are just spelling variations of the same word, then we also merged them into a single entry which still retains the variations. The final result is that now we have a lexicon with 64 877 lemmas in an uniform format.

3. Bulgarian Lexicon

The situation with the resources for Bulgarian is more difficult. There are a number of papers reporting the creation of large morphological or translation dictionaries for Bulgarian (see for instance Dimitrova et al. (1998), Paskaleva et al. (1993) and Koeva and Genov (2004)) but none of them has open source license. In fact we have not even seen this resources since they are inaccessible. The only free morphological dictionary is the one that is the basis of the Bulgarian spell checker in Open Office[2]. It contains 53 192 lemmas where each lemma is annotated with a paradigm number from the classification in Krustev (1984). Since this paradigms were already implemented in GF (Angelov, 2008), we just had to convert the resource to a format similar to the one used for English:[3]

```
kniga_N = mkN041 "kniga"
```

Here 041 is the number of the paradigm and `kniga_N` is the lemma identifier. We have used the same convention as in the English lexicon – the lemma identifier is the base form of the word transliterated in Latin followed by underscore and the part of speech tag.

Both the English and the Bulgarian lexicons are using the same part of speech tags. This is possible because the tag is

[2] This one is not even mentioned in the scientific literature.

[3] In the article we use transliteration while the actual resource is in Cyrillic script.

assigned to the lemma identifier and not to individual word forms. If we were to assign tags to the word forms instead, then we would necessarily need a different tag set for each language. For instance English has a definite article while in Bulgarian the definiteness is marked by inflecting the word. This means that we would need a different number of tags for each language. On the contrary if we assign the tag to the lemma identifier then we could reuse the same tag sets, but then each lemma identifier in the dictionary would correspond to an inflection table of a different size. This is also indicated by the use of the function `mkN041` which is defined in the Bulgarian grammar and is completely different from `mkN` that was used for English.

In general, each morphological paradigm for Bulgarian is implemented with a function which takes the base form as input and produces the inflection table for the word. For nouns the function also infers the gender.

4. Translation Dictionary

Once we got the two monolingual lexicons we had to merge them into a translation dictionary. We found four resources that are useful for linking but none of them is perfect.

The starting point was the English-Macedonian dictionary in Apertium (Forcada et al., 2011) which contains approximately 10 000 translation pairs. Bulgarian and Macedonian are closely related languages and most of the words can be translated from one of the languages to the other by applying a set of simple orthographic rules. Whenever the application of a rule to a word in the Appertium dictionary leads to a word from the monolingual Bulgarian lexicon, we get a candidate for English-Bulgarian translation pair. In this way, we managed to match most of the Macedonian words with Bulgarian words. However, the process is error-prone and we had to manually inspect and correct the mapping.

The second source is the Universal WordNet (de Melo and Weikum, 2009) which contains automatically learned WordNets for many languages including Bulgarian. Although there is existing Bulgarian WordNet (Koeva and Genov, 2004), it is not freely available and we did not use it. The manual inspection of the Universal WordNet showed that the quality is rather good but still we found many errors and some of them are probably still unnoticed. In contrast the Apertium dictionary was manually engineered and more trustworthy. When we merged the two dictionaries we kept the data coming from Apertium intact, and we only added translations for the words that we did not have already.

The third source is the popular English-Bulgarian dictionary KBEDict[4]. KBEDict is an open-source dictionary meant for language learners, which was build primarily by scanning paper dictionaries. Unfortunately it is very difficult to use it for computational purposes. It is basically a mapping from a word to a short text which lists the possible translations and their meaning. Although the text has a semi-structured form, there are still a lot of inconsistencies. There are also many errors left from the OCR software. For instance all Cyrillic letters which have the same or similar shape to some of the Latin letters are recognized as Latin. Similarly dot is often confused with comma and colon with semicolon. We wrote a script which parses as best as possible the English to Bulgarian part of the dictionary and produces, a more structured XML representation where each translation is clearly marked. We also used different heuristics to fix the errors from the OCR software. Furthermore, when we merged the new translation pairs with the already existing data, we checked that the Bulgarian word that we are about the add exists in the monolingual lexicon. This means that we can be sure that no OCR errors have slipped through.

Finally, we used EuroParl (Koehn, 2005) and automatic word alignment (Och and Ney, 2000) to get more data. In this case, we had to be even more careful since the word alignment produces a lot of noise in addition to useful data. Part of the problem is that the manual translation is almost never literal and some potential translations may not be translations at all out of the context. For that reason for each English word for which we still do not have translation we looked up only the most probable translation in EuroParl. The hope is that even if there are mistakes, they should be less frequent than the most probable translation. Of course this also means that we can miss alternative translations which are legitimate but just happen to be more rare. Again since the Word alignment is less reliable we used it to only add words which we did not get in other ways.

Before doing the actual linking we had to consider some specifics of the Bulgarian morphology.

First of all, the verbs in Bulgarian are classified by their perfective and imperfective lexical aspect. Roughly this corresponds to the difference between continuous and simple tenses in English. Since we want a translation dictionary, for every English verb, we had to match a pair of two Bulgarian verbs – one with perfective and one with imperfective aspect. The imperfective verb is formed from the perfective one by adding certain prefixes and suffixes while the stem of the verb remains the same. Not all verbs differ in aspect but when they do, using the wrong aspect is a serious error which either changes the meaning of the sentence or makes it ungrammatical.

The original monolingual Bulgarian lexicon contained a raw list of verbs where the aspectual pairs were not linked together. We searched for matching pairs of verbs by applying common derivational patterns while relying on the common stem. Whenever we found a pair, we merged the corresponding entries from the monolingual lexicon into one. If there is no match then we keep an entry which contains only one verb. After that we based the translation dictionary on the monolingual dictionary with merged entries rather than on the original source. The result looks like this:

```
admit_V =
        dualV (mkV186 "priznavam")
              (mkV161 "priznaja")
abdicate_V =
        singleV (mkV186 "abdikiram")
```

Here the English verb `admit` corresponds to two Bulgarian verbs `priznavam` and `priznaja` with different aspects. At the same time, there is only one translation for `abdicate` since the Bulgarian verb `abdikiram` can be

[4] http://kbedic.sourceforge.net/

used with both perfective and imperfective aspect. `dualV` is a function in the Bulgarian grammar which builds a bigger table from the tables of the two verbs. Similarly `singleV` builds a bigger table by copying the table for the single verb twice.
Furthermore, Bulgarian has medial and phrasal verbs which must always be used with the passive voice particle se/si or with a clitic pronoun. This is one more feature that was missing in the monolingual lexicon and we had to add it manually in the translation dictionary. Just replacing the English verb with the Bulgarian one plus the required particles and clitics is likely to result in incorrect sentence because they interact in a complex way with the syntax. The existing Bulgarian grammar takes care of the right placement, but whether the particles are needed at all can be determined only from the dictionary since English has none of this features.
Some extra work was needed for the nouns as well. In English it is very common to modify a noun with another noun. For example we say "computer music", but in Bulgarian we have to replace *computer* with an adjective which is morphologically derived from the noun. In the dictionary all nouns that have corresponding adjectives must be grouped with them. Again we did the mapping by developing a set of derivational patterns and by manual verification. Whenever we have a noun–adjective pair then we generate an entry like:

```
computer_N = dualN (mkN009 "kompjutar")
                   (mkA079 "kompjutaren")
```

where `dualN` is a function which glues the inflection tables for the noun and for the adjective in a bigger table. If there is no corresponding adjective then we simply produce:

```
boomerang_N = mkN007 "bumerang"
```

i.e. the same entry as in the monolingual lexicon except that the Bulgarian lemma identifier is replaced with the corresponding English identifier.
If we have a regular adjective which is not derived from a noun then it gets its own entry:

```
beautiful_A = mkA076 "hubav"
```

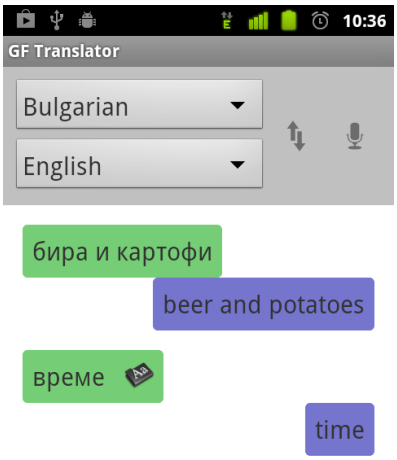
here again we have the same definition as in the monolingual lexicon except that the English lemma identifier is used with a Bulgarian word.
The final result is that our cleaned up dictionary currently contains about 27 000 translation pairs. Although Apertium already had 10 000 translation pairs, after the merge of pairs of verbs and of nouns and derived adjectives, this number was considerably reduced. We got more entries from the Universal WordNet, KBEDict and EuroParl. Of the 27 000 final pairs we have manually inspected and validated 11 000 pairs, i.e. 40% of the lexicon. In the process we also added some frequent but missing words by hand.
We also matched the words in the dictionary with the words in Penn Treebank and we found that there is 88% chance to find a translation for a random word in the corpus. This means that we already have translations for the most frequent words.
Note that we used Penn Treebank to extract information for the verb valency in English and we stored the information in the dictionary. We do not have the same information for Bulgarian but we just propagated the same valency from English to the corresponding verb in Bulgarian. Of course the valencies between English and Bulgarian does not always match. For instance the English verb may require a preposition, while the Bulgarian one does not or vice versa. However, that fact that a verb is transitive or not, and generally the number and the type of its arguments is a semantic property and generally propagates rather well from one language to another. Sometimes the same English verb with a different valency pattern may actually correspond to a different Bulgarian verb. Some of those discrepancies we will have to fix later.
This also opens the question for polysemous words. In our case we care only if the ambiguity leads to a different translation. In those situations we generate lemma identifiers such as `orange_1_N` (fruit) and `orange_2_N` (color). We match them with the same words in English but with different words in Bulgarian. Unfortunately currently there are very few such words and for the rest we either have only the most frequent translation or all possible translations listed as variants. For example if `orange_N` was not split into two entries it would look like:

```
orange_N = variants {mkN007 "portokal";
                     mkN054 "oranzevo"}
```

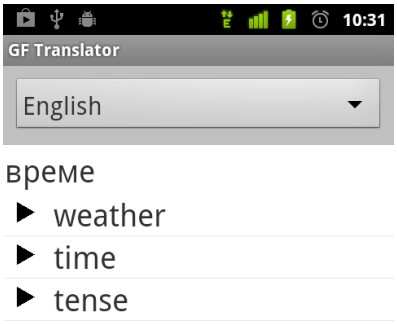
Here the keyword `variants` indicates that there are two possible realizations of the lemma `orange_N`. In English the variants are used only for introducing different spelling variations of the same word. In Bulgarian, however, they are used both for introducing spelling variations and for different senses. We already fixed some of this issues but it requires more work to complete it. This would require the development of a WordNet-like resource for Bulgarian.
A special instance of polysemy are nouns that indicate professions or other human roles. In Bulgarian they often come in pairs where one of them refers to a male and the other to a female role. The same distinction is also common in French and German, but is rare for instance in English and Swedish. In this case we again split the English lemma identifier into two identifiers corresponding to each case. For instance instead of having `teacher_N`, we have `teacherMasc_N` and `teacherFem_N`. Currently we have about 400 such pairs. The gender distinction is marked in English as well as in Bulgarian. In Bulgarian this is absolutely essential feature for correct translation and right gender agreement. In English it is needed only when the grammar has to decide between using `herself` or `himself` in reflexives and for deciding between `who` and `which`. In the later case the actual gender is irrelevant but the fact that it is specified indicates that the noun is not in neutr gender.

5. A Prototype for Translation System

As a front-end to the lexicon, we built an Android application (Figure 1, Angelov et al. (2014)) where the lexicon can be used off-line. It has two modes – in the first mode (Figure 1a) the user can use speech input to say a



(a) Translation Mode



(b) Dictionary Lookup

Figure 1: Android Mobile Frontend to the Dictionary

sentence, which is then recognized by Android's built-in speech recognizer and is parsed with the large GF grammar for either English or Bulgarian. Since the grammars are highly ambiguous we have used statistical disambiguation model trained on Penn Treebank (Angelov, 2011). The disambiguation model actually works on the level of the abstract syntax which is common for all grammars in the GF library. This means that although the model is trained on an English corpus, it could just as well be used for parsing other languages. In this case we use it for disambiguation in both English and Bulgarian. The translation itself is a matter of parsing the input sentence into an abstract syntax expression and then linearizing the same expression in the target language. The translated sentence is also pronounced by using Android's TTS service. Our continuous testing have shown that the application is already useful as a speech to speech translator for phrasebook kind of sentences. For large and complex sentences we usually either get an error from the speech recognizer, a wrong word sense in the translation, or just the parsing becomes too slow for running on the phone. Despite this we have found the interface very handy for testing the lexicon since the application is always available in our pockets.

If the user have said only a single word from the lexicon then next to the word we show a small icon which the user can click to navigate to the second mode (Figure 1b), where he/she can directly see all possible translations for this word. Clicking on the translation shows the full form inflection table, inherited parameters like noun gender, the verb type (i.e. medial or phrasal verb for Bulgarian), as well as the verb valency. The application will be published on the Android market in the near future.

6. Conclusion

The lexical resources that we have bootstrapped are available on-line at:

```
www.grammaticalframework.org/lib/src/
  english/
    DictEng.gf            -- monolingual
  bulgarian/
    DictBul.gf            -- monolingual
  translator/
    DictionaryEng.gf      -- translation
    DictionaryBul.gf      -- translation
```

GF provides APIs for Java, Python, C and Haskell which allow these lexicons to be accessed from other applications or exported to other formats.

We realize that although our monolingual lexicons are of considerable size, only about half of the words in them are connected with corresponding words in the other language. However, we considered that, as a first step, it is important to reuse as much as possible the available open-source resources as a basis for future extensions. Our research group has started parallel initiative for building similar dictionaries for Chinese, Finnish, French, German, Hindi, Italian, Urdu, Spanish and Swedish. Since they are connected to the same English lexicon we are actually already able to translate between Bulgarian and any of those languages. In fact the same Android application that we used for testing the English-Bulgarian dictionary can be used for translation between any of the mentioned languages. However, since all this dictionaries are aligned only to English it is possible that some sense distinctions are lost when going from one language to another via English. Testing this would require more work and fluent speakers in the corresponding language pair.

7. References

Krasimir Angelov, Aarne Ranta, and Björn Bringert. 2014. Speech-enabled hybrid multilingual translation for mobile devices. In *European Chapter of the Association for Computational Linguistics*, Gothenburg.

Krasimir Angelov. 2008. Type-Theoretical Bulgarian Grammar. In *Advances in Natural Language Processing*, pages 52–64. Springer-Verlag.

Krasimir Angelov. 2011. *The Mechanics of the Grammatical Framework*. Ph.D. thesis, Chalmers University of Technology.

Dana Dannells and Normunds Gruzitis. 2014. Extracting a bilingual semantic grammar from framenet-annotated corpora. In *Language Resources and Evaluation Conference*, Reykjavik, Iceland, May.

Gerard de Melo and Gerhard Weikum. 2009. Towards a universal wordnet by learning from combined evidence. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM 2009)*, pages 513–522, New York, NY, USA. ACM.

Grégoire Détrez and Aarne Ranta. 2012. Smart paradigms and the predictability and complexity of inflectional morphology. In Walter Daelemans, Mirella Lapata, and Lluís Màrquez, editors, *EACL*, pages 645–653. The Association for Computer Linguistics.

Ludmila Dimitrova, Nancy Ide, Vladimir Petkevic, Tomaz Erjavec, Heiki Jan Kaalep, and Dan Tufis. 1998. Multext-east: Parallel and comparable corpora and lexicons for six central and eastern european languages. In *IN PROCEEDINGS OF COLING*, pages 315–319. Addison-Wesley.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Mikel L. Forcada, Mireia Ginesti-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, JuanAntonio Perez-Ortiz, Felipe Sanchez-Martinez, Gema Ramirez-Sanchez, and FrancisM. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.

Albert Hornby. 1974. *Oxford Advanced Learner's Dictionary*. Oxford University Press, 3rd edition.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.

G. Totkov Koeva, S. and A. Genov. 2004. Towards bulgarian wordnet. *Romanian Journal of Information Science and Technology*, 7(1-2):45–61.

Borimir Krustev. 1984. *The Bulgarian Morphology in 187 type tables*. NI, Sofia, Bulgaria.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330, June.

R Mitton. 1986. A partial dictionary of English in computer-usable form. *Literary & Linguistic Computing*, 1(4):214–215, December.

F. J. Och and H. Ney. 2000. Improved statistical alignment models. In *ACL00*, pages 440–447, Hongkong, China, October.

Elena Paskaleva, Kiril Simov, Mariana Damova, and Milena Slavcheva. 1993. The long journey from the core to the real size of a large ldb. In *In Proceedings of ACL Workshop: Acquisition of Lexical Knowledge from Text*, pages 161–169.

Aarne Ranta. 2009. The GF resource grammar library. *Linguistic Issues in Language Technology*.

Aarne Ranta. 2011. *Grammatical Framework: Programming with Multilingual Grammars*. CSLI Publications, Stanford. ISBN-10: 1-57586-626-9 (Paper), 1-57586-627-7 (Cloth).

Josef Ruppenhofer, Michael Ellsworth, Miriam R. L. Petruck, Christopher R. Johnson, and Jan Scheffczyk. 2005. FrameNet II: Extended theory and practice. Technical report, ICSI.