Sharing resources between free/open-source rule-based machine translation systems: Grammatical Framework and Apertium

Grégoire Détrez[†], **Víctor M. Sánchez-Cartagena**[*‡], **Aarne Ranta**[†]
gregoire.detrez@gu.se, vmsanchez@prompsit.com, aarne@chalmers.se
[†]Department of Computer Science and Engineering, Gothenburg University,
SE-405 30 Gothenburg, Sweden
[*]Dep. de Llenguatges i Sistemes Informàtics, Universitat d'Alacant,
E-03071, Alacant, Spain
[‡]Prompsit Language Engineering,
Av. Universitat, s/n. Edifici Quorum III. E-03202 Elx, Spain

Abstract

In this paper, we describe two methods developed for sharing linguistic data between two free and open source rule based machine translation systems: Apertium, a shallow-transfer system; and Grammatical Framework (GF), which performs a deeper syntactic transfer. In the first method, we describe the conversion of lexical data from Apertium to GF, while in the second one we automatically extract Apertium shallow-transfer rules from a GF bilingual grammar. We evaluated the resulting systems in a English-Spanish translation context, and results showed the usefulness of the resource sharing and confirmed the a-priori strong and weak points of the systems involved.

Keywords: rule-based machine translation, linguistic resource sharing, open-source linguistic resources

1. Introduction

Machine Translation (MT) can be defined as the use of software to translate content from one natural language, the source language (SL), into another, the target language (TL). Two main MT paradigms can be established according to the kind of knowledge involved in the translation process.

On the one hand, corpus-based approaches use large parallel corpora as the source of knowledge. A parallel corpus is a collection of parallel texts, that is, texts in one language together with their translation into another language. The statistical machine translation (SMT; Koehn (2010)) corpus-based approach is currently the leading paradigm in MT. SMT systems can be built with little human effort, provided that a large enough parallel corpus is available.

Rule-based machine translation (RBMT; Hutchins and Somers (1992)) systems on the other hand are best characterized by their use of explicit linguistic knowledge. This knowledge can take many forms, from simple monolingual dictionaries to complex semantic structures but it usually needs to be manually encoded by experts, which represents a big part of the effort needed to create such systems and has a great influence on their overall performance.

Among the different RBMT approaches, transfer-based systems are those in which the translation process can be split in the following three steps: they perform an analysis of the SL text into an SL intermediate representation; after that, the intermediate representation is transferred to the TL; and finally the translation is generated from the TL intermediate representation.

Software licensed as Open Source allows anyone to study, change and distribute the software to anyone and for any purpose. In the case of RBMT systems, this creates the possibility of reusing the linguistic knowledge encoded in one system to create, or at least bootstrap, a different system.

In this paper, we started exploring the many possible ways for sharing linguistic data between the free/open-source RBMT systems Apertium (Forcada et al., 2011) and Grammatical Framework (GF, Ranta (2011)) with two new methods. Apertium is a shallow-transfer system, which means that it does not perform a full syntactic analysis to build the intermediate representation. Contrarily, GF is a multilingual grammar formalism that has been used to build MT systems, among other applications. The methods we developed allowed us to create two RBMT systems from the same resources and do an empirical comparison between Apertium and GF, analyzing their strong and weak points.

Previous strategies to share Apertium or GF linguistic resources include using Apertium data to enrich statistical machine translation (Tyers, 2009; Sánchez-Cartagena et al., 2011) and example-based systems (Sánchez-Martínez et al., 2009), and combining SMT systems with GF (Enache et al., 2012). Resource sharing between Apertium and GF has however never been explored.

2. Integration

We have developed two sharing strategies: augmenting the GF lexicon with entries from an Apertium dictionary, and creating Apertium shallow-transfer rules from GF grammars. They are described in this section together with the main differences between GF and Apertium.

2.1. Differences between GF and Apertium

GF (Grammatical Framework, Ranta (2011)) is a multilingual grammatical formalism, and the key point of its design is the separation of a language independent abstract syntax from multiple concrete syntaxes. GF also provides the Resource Grammar Library (RGL; Ranta (2009)), a model of the low level structures of syntax and morphology for (at the time of this writing) 29 natural languages. Thanks to the RGL, When writing a domain-specific grammar, one needs only to concentrate on the abstract syntax for her domain, leaving the tedious linguistic details to the library. However, the linguistic information of the RGL can also be exploited to perform open-domain translation by using the common API of the RGL as a pivot (Figure 1), which is the configuration explored in this paper. In a taxonomy of MT according to the abstraction level of the intermediate representation, this last configuration can be seen as a form of syntactic transfer.

Unlike GF, the Apertium shallow-transfer RBMT platform (Forcada et al., 2011) was initially designed for open-domain translation. Apertium uses a simple, flat, intermediate representation: a sequence of *lexical forms* representing the lemma, lexical category and morphological inflection information of the words to be translated. E.g.:

> *the*.**det**.def *red*.**adj** *car*.**n**.pl

The translation between the source-language (SL) and the target-language (TL) lexical forms is carried out by a set of shallow-transfer rules performing operations such as agreements, re-orderings, preposition changes, etc (see Figure 2). Each rule processes a chunk of lexical forms and they are applied in a greedy manner.

While GF guarantees a grammatically correct output and allows more sophisticated transformations (e.g. long-distance re-orderings), the quality of the translation drops when the sentence cannot be fully parsed because out-of-vocabulary words or an irregular grammatical structure. Although, for such sentences, GF generates some partial subtrees (Angelov, 2011), the shallow-transfer approach followed by Apertium allows for more robustness.

The 37 language pairs supported by Apertium include 17 languages not present in GF RGL (Aragonese, Asturian, Basque, Breton, Galician, Icelandic, Indonesian, Kazakh, Macedonian, Mataysian, North, Nynorsk, Occitan, Portugese, Serbo-Croatian, Slovenian, Sámi, Tatar and Welsh[1]), while the RGL contains data for 23 languages (Amahric, Chinese, Estonian, Finnish, German, Greek, Hebrew, Hindi, Japanese, Latin, Latvian, Mongolian, Nepali, Persian, Punjabi, Polish, Russian, Sundhi, Swahili, Thai, Tswana, Turkish and Urdu[2]) and more than 700 language pairs not yet present in Apertium. This shows the potential advantages of sharing resources between Apertium and GF.

2.2. Augmenting the GF lexicon with Apertium data

Lexica in RBMT contain the analysis of each word the system is able to translate or generate, and mappings between analyzed forms in different languages. In Apertium and GF, inflection paradigms are used to efficiently encode them.

The first system in our comparison was based on the GF Resource Grammar Library, in which lexicon entries from open lexical categories have been replaced with the information from the Apertium dictionaries. Although the GF Resource Grammar Library already contained a huge English lexicon, for the purposes of this work we only included in the resulting system the entries from closed lexical categories. Regarding Spanish, the GF lexicon contains all the words from closed lexical categories but only a few words from open categories. The latter were removed too and replaced with the ones from Apertium.

For many lexicon entries, porting them from Apertium to GF simply meant dealing with the different encoding details of both systems. First, we expanded the Apertium SL monolingual dictionary entries (i.e., to apply the corresponding paradigm to the stem to generate all word forms) and created a new entry in the GF SL lexicon for each of them by providing to a smart paradigm (Détrez and Ranta, 2012) all the expanded forms.

For instance, the following was the entry for the English noun *car* in the Apertium monolingual dictionary.

```
<e><i>car</i><par n="house__n"/></e>
```

The entry indicates that the noun *car* is inflected in the same way *house* is (the plural form is built by adding -s, and when adding the genitive marker to it, the suffix becomes -s'.) In that case, the result of the expension was:

```
car:car.n.sg
cars:car.n.pl
car's:car.n.sg.gen
cars':car.n.pl.gen
```

And the resulting GF entry, using the smart paradigm mkN for nouns:

```
lin car_N = mkN "car" "cars"
                "car's" "cars''" ;
```

The same process was repeated for the target language. In our example, the entry for the translated lexeme in the target language was:

```
<e><i>coche</i><par n="abismo__n"/></e>
```

Which was converted to:

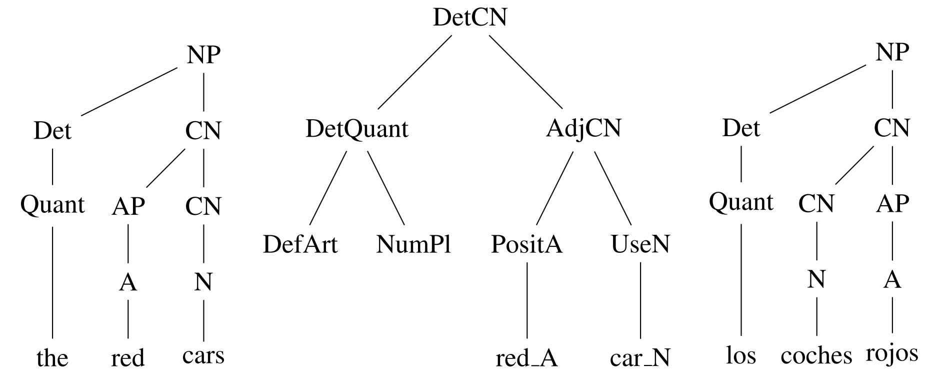
```
lin car_N = mkN "coche" "coches"
                masculine ;
```

Note that in both cases the GF function is named *car_N* instead of *coche_N* in the target language. This was necessary to map the entry to its source language equivalent and it was achieved by looking up the lemma in Apertium's bilingual dictionary.

This simple startegy was made possible by the design of GF's smart paradigms (Détrez and Ranta, 2012) which allowed the creation of a valid lexicon entry giving only partial information, the missing forms and parameters being infered using the language morphology. For instance, adjective entries in the English GF lexicon had their adverbial form attached, while Apertium had separated entries for adjectives and adverbs. When porting English adjectives from Apertium to GF, we let the GF smart paradigms

[1] List of stable language pairs. Retrieved October 21, 2013, from http://wiki.apertium.org/wiki/Main_Page.

[2] The Status of the GF Resource Grammar Library. Retrieved October 21, 2013, from http://www.grammaticalframework.org/lib/doc/status.html.



(a) English concrete tree obtained after parsing *the red cars*

(b) Abstract syntax tree

(c) Spanish concrete tree which is linearized to *los coches rojos*

Figure 1: Example of parse trees in GF when performing an English-Spanish open-domain translation

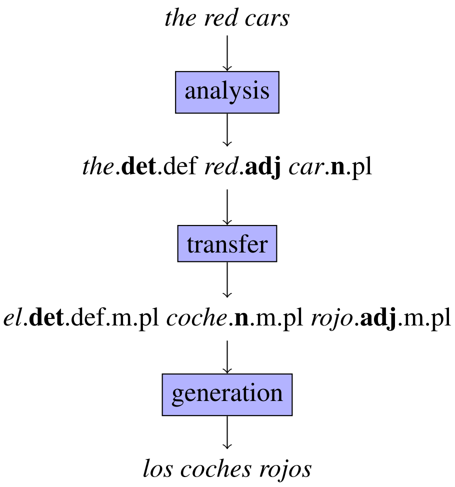

the red cars
analysis
the.det.def red.adj car.n.pl
transfer
el.det.def.m.pl coche.n.m.pl rojo.adj.m.pl
generation
los coches rojos

Figure 2: Lexical forms produced by the Apertium engine when translating the English phrase *the red cars* into Spanish. Lemmas are shown in italics and lexical categories in bold. *det* stands for determiner, *adj* means adjective and *n* means noun. The determiner is definite (*def*), the gender is masculine (*m*) and the noun is plural (*pl*).

infer the adverbial form of the adjective. An other example, still regarding adjectives: Spanish adjectives are usually placed after the noun they modify, but a few of them, called prepositive adjectives, are placed after the noun. This feature was needed in the GF lexicon, but was not present in the Apertium one. As a consequence, when including Spanish adjectives from Apertium in the GF lexicon, we could not provide any information about this to the smart paradigms. When there was not enough information, the smart paradigms chose the most common option: in this case, that the adjective was not prepositive. Finally, English nouns contain a humanity feature in GF, which is not encoded in Apertium and in this case, all Apertium nouns were imported as non-human (the most common value).

In addition, since GF uses a deeper intermediate representation, some additional linguistic information was required when inserting certain entries in the GF lexicon. In particular, in the case of verbs, the GF lexicon contains valency information used in parsing. For instance, the valency V indicates an intransitive verb (for example: *run*), V2 a transitive verb (*hit*), VA states that a verb is complemented by an adjective (*become*) and so on. Since it was not possible to infer the verb valencies we imported them from the existing GF English lexicon[3] and used them for both English and Spanish verbs (in the RGL API, the valency is encoded in the language-independent abstract syntax, so it is necessarily the same for linearization of the same abstract function.)

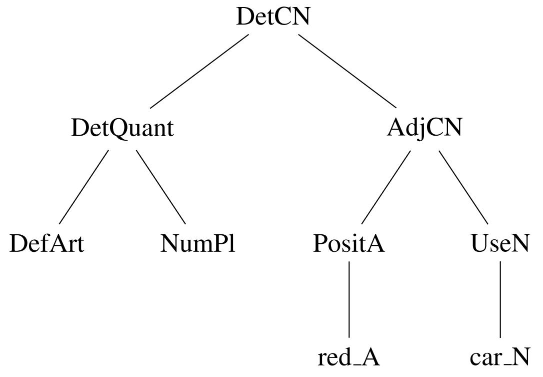
2.3. Generating Apertium shallow-transfer rules from GF data

The second system used the Apertium engine and lexicon but we extracted structural transfer rules from the GF resource grammar library. The Apertium shallow-transfer rules process fixed-length chunks of lexical forms and perform agreements, re-orderings, preposition changes and other grammatical transformations. Naturally, since Apertium does not perform a full parsing, we could not simply re-encode the gf grammars into Apertium rules. Instead, we developed a method based on flattening abstract syntax trees.

In a nutshell, our strategy involved generating, for each GF abstract syntax function from the Resource Grammar Library, all the possible abstract trees which could be built (up to a certain depth). Each tree was then linearized in both SL and TL to obtain a bilingual phrase (the GF engine provides the word-by-word alignment), and an Apertium shallow-transfer rule was extracted from the pair of linearizations using the algorithm developed by Sánchez-Martínez and Forcada (Sánchez-Martínez and Forcada, 2009).

The depth limit was important to obtain a finite and manageable set of abstract syntax trees. In order to avoid the generation of an unmanageable set of bilingual phrases, we also took advantage of the fact that in GF the grammar rules are not influenced by a word form but only by the features (e.g. two masculine nouns will appear in exactly the same

[3] The GF RGL originally contains over 60.000 entries in the English lexicon, but only a few dozens in the Spanish one.



(a) GF abstract syntax tree

the.**det**.def *red*.**adj** *car*.**n**.pl
→ *el*.**det**.def.m.pl *coche*.**n**.m.pl *rojo*.**adj**.m.pl

(b) its linearization into English (left) and Spanish (right) when replacing the linearization of terminal symbols with Apertium lexical forms

the.**det**.def **adj** **n**.pl|m.pl
→ *el*.**det**.def.m.pl **n**.m.pl **adj**.m.pl

(c) Same example with nouns and adjectives replaced by word classes.

the.**det**.def **adj** **n**.pl|m.pl
→ *el*.**det**.def.m.pl $3.**n**.m.pl $2.**adj**.m.pl

(d) Apertium rule extracted from it. The rule matches the definite determiner *the*, followed by any adjective and a plural noun whose gender after being looked up in the bilingual lexicon is masculine, and its number is plural.The expression $i means that the lemma is obtained by looking up in the bilingual lexicon the i-th matching SL lexical form

Figure 3: Steps carried out obtain an Apertium shallow transfer rule from a GF abstract syntax tree.

set of trees) Thus, for each open lexical category, only one for each combination of SL and TL features was included in the GF lexicon used to generate bilingual phrases. For instance, with regard to common nouns when translating from English to Spanish, the only relevant feature for translation was the gender in Spanish. Consequently, we only needed to include in the lexicon a noun which is masculine in Spanish (such as *car)*, and another one which is feminine (for instance, *house*). In addition, other modifications were carried out to ensure that Apertium shallow-transfer rules could be obtained from the pair of linearizations. First, the linearization of each GF lexical function was replaced by its corresponding Apertium lexical form. In this way, pairs of lexical form sequences were obtained when linearizing the abstract function trees. These pairs could then be directly converted into Apertium shallow-transfer rules.

However, the approach which has just been described would generate a vast amount of rules, since a rule for each combination of lexical entries would be obtained. Since it was desirable to obtain a smaller set of rules, we performed a further modification: the introduction of word classes. We modified again the GF lexicon, in which previously the original entries have been replaced with Apertium lexical forms, and replaced lexical forms from open lexical categories with word classes. The word class of an SL lexical form is defined as the concatenation of its lexical category, morphological inflection information and morphological inflection information obtained when looking it up in the Apertium bilingual dictionary. Word classes group together words which behave in the same way when being translated. For example, the word class of lexical forms such as *car*.**n**.pl, *phone*.**n**.pl, or *day*.**n**.pl is **n**.pl—**m**.pl, since all of them are plural nouns in English which are translated as masculine plural nouns in Spanish. Word classes of TL lexical forms only contain the lexical category and TL morphological inflection information (the bilingual dictionary is not involved). Figure 3c shows the pair of linearizations from the tree presented in figure 3a, in which lexical forms have been replaced by word classes.

Although the pair of lexical form sequences just shown is more similar to an actual Apertium rule than the previous examples, one detail remain to be fixed: alignments were needed in order match SL and TL word classes and allow the Apertium engine to collect the lemmas of the TL word classes by looking up in the bilingual dictionary the corresponding SL words. Fortunately, the GF engine provides them. The final Apertium rule obtained is depicted in figure 3d. This rule matches the definite determiner *the*, followed by an adjective and a plural noun which is masculine and plural in Spanish, and generates, in Spanish, a definite, masculine, plural determiner; a masculine plural noun whose lemma is obtained by looking up in the bilingual dictionary the lemma of the English noun, and a masculine plural adjective whose lemma is obtained by looking up in the bilingual dictionary the lemma of the English adjective.

3. Evaluation

We used the methods described above to build two English-Spanish MT systems stemming from the same resources: the Apertium lexicon and the GF RGL. *sharedApertium* is an Apertium-based system containing the original Apertium lexicon and a set of shallow-transfer rules created from the GF RGL, while *sharedGF* is a GF-based system in which the lexicon has been ported from Apertium.

We performed an automatic evaluation using two subsets of the *newstest2011*[4] set. We computed BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005) and TER (Snover et al., 2006) scores for the aforementioned systems, along with out-of-the-box Apertium and a word-for-word translation with the Apertium lexicon. The subset *newstest2011A* (1896 sentences) contains the parallel sentences from newstest2011 which can be parsed (either fully or partially) by GF in a reasonable time, while *newstest2011B* (130 sentences) contains only those fully parsed by GF. Results are shown in Table 1.

[4] Distributed as part of the WMT 2011 shared translation task: http://www.statmt.org/wmt11/translation-task.html

Corpus	System	BLEU	METEOR	TER
newstest2011A	*sharedGF*	0.027	0.181	0.847
	sharedApertium	**0.138**	**0.390**	**0.678**
	Apertium word-for-word	0.111	0.368	0.703
	Apertium	0.200	0.443	0.617
newstest2011B	*sharedGF*	0.152	0.388	0.703
	sharedApertium	0.148	0.391	0.691
	Apertium word-for-word	0.106	0.361	0.713
	Apertium	0.212	0.451	0.620

Table 1: Values of the evaluation metrics obtained by the different English–Spanish MT systems. A score in bold for *sharedApertium* means that it outperforms *sharedGF* by a statistically significant margin computed by paired bootstrap resampling (Koehn, 2004) with $p = 0.05$. An underlined score indicate that the system outperforms Apertium word-by-word translatio, according to the same criterion.

The most remarkable conclusion that can be drawn from the results is that our resource sharing strategies eased the development of new RBMT systems: an Apertium-based system which outperformed word-for-word translation has been created without manually writing a single shallow-transfer rule; and a GF-based system, which also outperformed Apertium word-for-word translation on the smallest corpus, has been built despite that the GF lexicon only contained originally a few entries in the Spanish side.

Regarding the differences between *sharedGF* and *sharedApertium*, GF performed poorly on the bigger, *newstest2011A* corpus, mainly due to out-of-vocabulary words and out-of-grammar constructions. These are less of an issue for Apertium, which simply translates word-by-word when no rule matches the input chunks. An example of this situation is presented in Figure 4.

sharedGF catches up with *sharedApertium* on the smaller, fully-parsed, evaluation corpus. As pointed out previously, analyzing the whole sentences allowed GF to perform more accurate translations than Apertium for some constructions, as in the example presented in figure 5. The GF parser was also able to automatically detect named entities, which lead to the correct translation of Saxon genitives even when the proper noun was not in the lexicon. See Figure 6 for an example.

However, even when the sentence was fully parsed, the GF-based system had some drawbacks when compared to Apertium. For instance, the Apertium analyzer correctly handled most multi-word expressions encoded in the lexicon because it always tries to match the longest possible segments, but the GF parser relies on statistics to choose among the possible parse trees, which could lead to situations such as the one shown in Figure 7. One could remedy to this problem by tuning the probability in the GF grammar—either manually or using treebank data—so that the idiosyncratic interpretation is chosen over the compositional one. On the other hand, GF could also analyze discontinuous multiword expressions which cannot be encoded in Apertium's lexicon.

4. Conclusions and future work

We have presented two strategies for sharing linguistic resources between Apertium and GF and used them to create two RBMT systems stemming from the same linguistic resources. Our experiments showed the usefulness of the resource sharing and confirmed the a-priori strong and weak points of the systems involved.

Possible future works include exploring other ways to share Apertium and GF's resources. For instance, porting the GF lexicon to Apertium or using GF smart paradigms (Détrez and Ranta, 2012) to ease the creation of the Apertium lexicon. A deeper integration of Apertium and GF could also be achieved by combining them at runtime, following a approach similar to the strategy designed to integrate GF and SMT(Enache et al., 2012).

5. Acknowledgements

Work partially supported by the Spanish government through project TIN2012-32615 and by Generalitat Valenciana through grant ACIF/2010/174 and Swedish Research Council through financial support under grant nr. 2012-5746 (Reliable Multilingual Digital Communication: Methods and Applications).

6. References

Angelov, K. (2011). *The Mechanics of the Grammatical Framework*. Ph.D. thesis, Chalmers University Of Technology.

Banerjee, S. and Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. pages 65–72.

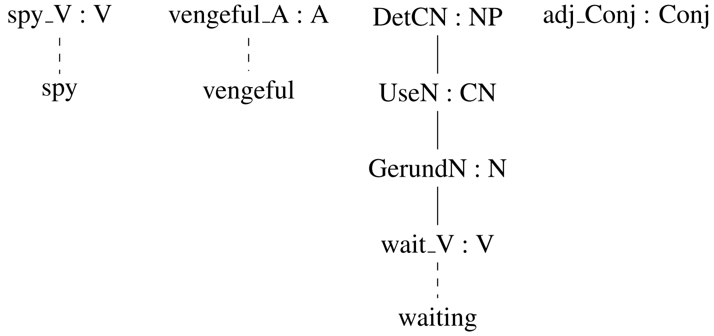
Détrez, G. and Ranta, A. (2012). Smart paradigms and the predictability and complexity of inflectional morphology. In *13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 645–653, Avignon, France.

Enache, R., España-Bonet, C., Ranta, A., and Màrquez, L. (2012). A hybrid system for patent translation. In *The 16th Annual Conference of the European Association for Machine Translation*, pages 269–276, Trento, Italy, May.

Forcada, M. L., Ginestí-Rosell, M., Nordfalk, J., O'Regan, J., Ortiz-Rojas, S., Pérez-Ortiz, J. A., Felipe Sánchez-Martínez, G. R.-S., and Tyers, F. M. (2011). Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144. Special Issue: Free/Open-Source Machine Translation.

Hutchins, W. and Somers, H. (1992). *An introduction to*

source: Vengeful **hackers** and spies **are** waiting



sharedGF:	vengativo espiar
sharedApertium:	Vengativo **hackers** y espías **son** esperando

Figure 4: SL sentence from the *newstest2011A* evaluation corpus, partial parse trees obtained by the GF parser of the *sharedGF* system, their translation, and the translation of the same SL sentence by *sharedApertium*. Observe that the out-of-vocabulary word *hackers* prevents GF from fully parsing the sentence.

source:	But the man remains (V) skeptic
sharedGF:	Pero el hombre queda (V) escéptico
sharedApertium:	Pero el hombre restos (N) escéptico
source:	The matter examines the type of damage (N)
sharedGF:	El asunto examina el tipo de daño (N)
sharedApertium:	El asunto examina el tipo de averiar (V)

Figure 5: SL sentences from the *newstest2011A* evaluation corpus and their translation with the systems being evaluated. Lexical category is shown in parentheses.

source:	This is Dan Brown's success mechanism
sharedGF:	Éste es mecanismo de éxito de Dan Brown
sharedApertium:	Esto es Dan Brown mecanismo de éxito

Figure 6: SL sentence from the *newstest2011B* evaluation corpus and its translation with the systems being evaluated.

source:	An extra portion of sugar is needed for lemon ice cream
sharedGF:	Una porción extra de azúcar está necesitada para crema de hielo de limón
sharedApertium:	Una porción extra de azúcar es necesitar para limón helado

Figure 7: SL sentence from the *newstest2011B* evaluation corpus and its translation with the systems being evaluated. The right translation into Spanish of the multi-word expression *ice cream* is *helado*, while *crema de hielo* is the literal translation.

machine translation, volume 362. Academic Press New York.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 4, pages 388–395.

Koehn, P. (2010). *Statistical Machine Translation*. Cambridge University Press, New York, NY, USA, 1st edition.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318.

Ranta, A. (2009). The GF resource grammar library. *Linguistic Issues in Language Technology*, 2(2), December.

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

Sánchez-Cartagena, V. M., Sánchez-Martínez, F., and Pérez-Ortiz, J. A. (2011). Integrating shallow-transfer rules into phrase-based statistical machine translation. In *Proceedings of the XIII Machine Translation Summit*, pages 562–569, Xiamen, China, September.

Sánchez-Martínez, F. and Forcada, M. L. (2009). Inferring shallow-transfer machine translation rules from small parallel corpora. *Journal of Artificial Intelligence Research*, 34(1):605–635.

Sánchez-Martínez, F., Forcada, M. L., and Way, A. (2009). Hybrid rule-based – example-based MT: Feeding apertium with sub-sentential translation units. In Forcada, M. L. and Way, A., editors, *Proceedings of the 3rd Workshop on Example-Based Machine Translation*, pages 11–18, Dublin, Ireland, November.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *In Proceedings of Asso-*

ciation for Machine Translation in the Americas, pages 223–231.

Tyers, F. M. (2009). Rule-based augmentation of training data in breton-french statistical machine translation. In Màrquez, L. and Somers, H., editors, *Proceedings of the 13th Annual Conference of the European Association for Machine Translation*, pages 213–217.