Towards a RB-SMT Hybrid System for Translating Patent Claims – Results and Perspectives

Ramona Enache and Adam Slaski

Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

1 Introduction

There is a growing interest for patents translation. The rapid advances in technology brought about a large number of inventions that were patented. This created a multilingual database of patented work from all over the world. Although the patents follow the same structural patterns, they are written in the language of the country where the patent is obtained, so it is difficult to get an overview of all the patents available in the collection.

In this context, it is necessary to be able to translate patents, for a better collaboration within the field - among scientists from different countries who need access to information about latest discoveries. Also there is the issue of patent acquisition, when it is necessary to search if a similar invention was patented before. In this case, it would be possible to access more patents - via translation and search for relevant information there with information retrieval techniques.

Because of the large amount of information available, human translation would be too costly and for this reason, it is necessary to use machine translation for this task.

Our approach to patents translation uses GF(Grammatical Framework)[1], a grammar formalism used for multilingual natural language applications. Its key concept is the division of a grammar in an abstract syntax part - semantic interlingua and concrete syntaxes - corresponding to target languages. The largest and most general example of such a grammar is the resource library [2], comprising 18 languages, for which the main grammatical constructions are provided. The library can be further used by domain-specific grammars, that can use the grammatical constructions from here in order to build syntactically correct constructions, thus alleviating the burden of handling linguistic difficulties and allowing a better focus on the higher-level details. In this way, one can concentrate better on the semantic structure of the grammar, and even without linguistic knowledge, can build a domain-specific grammar without much effort.

Because of the way the multilingual grammar is structured, it can also be used as a rule-based machine translation system between any pair of languages, for which a concrete syntax is provided. The task of translation is resumed to parsing from the source language to an abstract syntax tree and linearizing it in the target language. However, the system is restricted to the language generated by the grammar - a controlled language with limited vocabulary and limited set of constructions. We want to extend this system to one capable of handling free text - as it is the case with the patent claims. For this purpose, we want to enhance GF with SMT elements in order to build an RB-SMT hybrid system for translating patent claims. Two obvious directions to improve the GF baseline is to increase coverage by extending the lexicon on the fly and to make the parser robust in order to handle constructions that are not within the bounds of the grammar.

The current system is capable of English to French translation and it acts as following:

– builds automatically the lexicon(abstract syntax and English concrete syntax) from the English text - using a POS tagger for correct labelling of the entries and a lemmatizer for obtaining the base form
– builds the French concrete syntax for the lexicon using statistical tools for translation
– applies the GF translation mechanism on sentences that can be parsed, otherwise uses a chunker and translates parts of the sentence that can be handled by the grammar.

2 Related Work

Much work was done in the last decade in the field of patents translation between various pairs of languages. Various methods were used for this task, ranging from SMT [3] to hybrid systems [4], [5].

Moreover, various components associated to patents translation were treated separately - extraction of multilingual lexicon[6] and [7], natural language analysis of patent claims [8], handling long sentences in translation [9] and generating claims using machine translation [10].

In addition to this, an ongoing European project in patents translation, Pluto [11] is focused intensively on this task.

3 The Patents Translation System

3.1 Lexicon Building

As we mentioned before, a key issue in translation using a GF grammar is the limited lexicon. Since the vocabulary of patent claims if virtually unlimited, we need to build the lexicon for our grammar on the fly. For this purpose we need to construct an abstract syntax for the lexicon and concrete syntaxes corresponding to the languages between which we want to translate.

We use the GF library multilingual lexicon containing the most common entries for structural parts of speech - prepositions, conjunctions and pronouns. Consequently, we only need to take care of nouns, adjectives, verbs and adverbs. In order to identify them properly, we need a POS tagger. Further more, because an entry of the abstract syntax should generate all forms, we need to have the words lemmatised. For convenience, we created the abstract syntax for the lexicon from the English claims, because the language is morphologically simpler and there are more tools available for POS tagging and lemmatizing.

Regarding the choice of a POS-tagger, we decided upon GENIA [12] prepared specially to process texts from the biomedical domain. We made a previous attempt at this by using a general state-of-the-art POS tagger – Stanford POS tagger [13], but the results were visibly worse, because of incorrect classification of verbs and nouns. For example the noun "claim" was always classified as a verb. Also the adjective "human" was incorrectly classified as noun in all its occurences, which makes it more difficult to obtain a correct translation from such a lexicon. It is likely that the behaviour of the Stanford POS tagger would improve when training on a specific corpus, not on the general one that comes along with the distribution, but since there was no large annotated corpus from the biomedical patents domain, we could not follow this lead. The resulting POS tagged entries are further lemmatised with the GENIA lemmatizer.

Using this procedure, we built a lexicon of almost 700 entries based on the first 200 claims. The entries are nouns, adjectives, adverbs and verbs.

As expected, the precision of the lemmatiser is very high - over 90%, one still needs to go through the abstract syntax and check the entries, because every inconsistency introduced at this step can greatly influence the behaviour of the system later - by increasing the number of ambiguities, or failing to produce the right parse trees because of entries tagged incorrectly in the lexicon.

An example of such a bug may be recognition of a roman number one (*i*, used in claims for enumeration) as a noun. Plural form of such a “noun” is *is*, the same as the present form of *to be*; the reader may imagine number of ambiguities caused by this malfunction. Also, past forms of irregular verbs like *said*, were not properly lemmatized to the base form.

The next step after removing noise is to generate proper inflection. Due to simplicity of the English inflection this is rather straightforward, especially because scientific terms, like most words in the claims have a regular inflection. The GF library provides a large English dictionarry with almost 50,000 entries that can be used for getting the inflection forms of some of the words – all irregular verbs can be found here and properly inflected. For the words which don't exist in the dictionarry, we use the GF paradigms to build the inflection table from the base form. The assume a regular inflection scheme for the words, which proved to work for the words in our test lexicon.

The only problem when generating lexicon is to extract the correct valency for verbs. Unfortunately, inductive generation of a valency dictionary is a well known problem in computational linguistics. Therefore, did not attempt to solve the problem in general, but try to develop an ad-hoc solution with a simple heuristic. The heuristic is to assume that if all verbs are two-place verbs, as this turned out to be the case for the large majority of the verbs encountered in the 200 claims considered for initial evaluation. In case a verb needs to be used without an object, a coercion to simple verb is provided. We considered the situations when a verb is followed by:

1. a nominal phrase then the verb takes one direct object,
2. a proposition then the verb takes one propositional object,

Now the English lexicon is complete and suitable for parsing. This is time to construct lexicon in the target language. In our case the target language is French.

We considered two approaches to this problem. The first one would be to extract the translation of the English entries from the lexical tables generated by a SMT system trained on the patents corpus. However, here the problem was that the French words might not be lemmatised. The second solution was to use Google Translate [14], as a general statistical translation system and to trick it to generate the base forms of the French words, by feeding the correponding basic forms from the English lexicon. An evaluation on the French lexicon generated with this technique showed that although the French translations were adequate, they were not properly lemmatised in this case either. Almost 70% of the translated adjectives were in plural form. For nouns and adverbs the situation was not as dramatic, since one can obtain a noun in singular by asking for the translation of a noun phrase with the proper indefinite article, and adverbs have no inflection forms.

In order to overcome the lemmatization problem, we employed a morphological dictionary Morphalou [15]. Equipped with the information about part of speech of each word a simple script was able to find in Morphalou lemmas of French words. When words were correctly lemmatized all the inflection was provided with smart paradigms.

3.2 Preprocessing

Before grammar may be applied to the corpus in order to parse it, some preprocessing is necessary. The reason is that certain information from the text would not influence the translation, but just make the parser slower - like long numbers, proper names, abbreviations. Hence, we created a number of preprocessing scripts that replace named-entities with a place-holder name and all numbers with "1".

Moreover, since the claims belong to the biomedical domain, it is well-known that statistical tools cannot render good translations of chemical compounds. For this reason, we recognise compounds in the text, and translate them separately with a another grammar.

Compound Recognition and Translation By “compound name” we understand here a string defining a chemical formula. Consider the following examples:

1. cis-4-cyano-4-(3-(cylopentyloxy)-4-methoxyphenyl)cyclohexane-1-carboxylic acid
2. 3,4,5-trimethoxybenzoic acid,

3. carboxylic acid.

As we see compounds may consist of either one or more words. Some of words may contain digits, punctuation and parentheses, while other may look like ordinary words. In order to detect compound names of all those types we implemented a cascade of filters. Base filter is a list of known chemical terms. The list is obtained from [16] and other filters were written manually.

What makes the difference between a rule-based approach and a mere translation of each word from the compound is the presence of "functional words" like *acid, ester, aldehyde, etc.*, which change place in traslation, such as in the translation of the first compound: *acide cis-4-cyano-4-(3-(cyclopentyloxy)-4-méthoxyphényl) cyclohexane-1-carboxylique.*

For this reason, the grammar swaps the place of the radical and the functional word in translation. For simplicity's sake, the grammar doesn't aim at a more in-depth analysis of the compounds, but only of the details that would make a difference in translation.

Named Entity Recognition Regarding the named entity recognition step, consider the following example:

> monoclonal antibody of class IgG which is produced from hybridoma ATCC CRL 8001 (OKT3).

Strings *IgG* and *ATCC CRL 8001 (OKT3)* have a special function in the text above. They are the abbreviation of a substance, and we can find its translation in a dictionary or by using the lexical tables of a SMT system – since no lemmatization is necessary for proper names.

After applying the named entity recognizer script, the sentence would be transformed to:

> monoclonal antibody of class AA which is produced from hybridoma AA.

Because we need to restore the proper names in the translated text after postprocessing, we store the names in a file indexed by their occurence in the text, so the place holder name *AA* will also be indexed, in order to make the replacement more reliable, in case that some proper names could swap place in translation.

For the actual named entity recognizer script, the first approach was to use the Stanford POS-tagger and look for entries tagged as proper names. However, it turned out that a simple heuristic would work just as well. The heuristics is to consider proper names the words starting with capital letter(after lowercasing the sentences), or words containing numbers or special characters inside. A group of proper names, is further on merged into only one proper name for simplicity, same for proper names separated with hyphen, and proper names followed by a proper name between parantheses. This script lead to 100% precision and recall for 200 claims that we used as fresh testing corpus– where the proper names were manually annotated and the output was compared to that of the named entity recognizer. In this case 176 proper names were properly classified and replaced with the placeholder name.

3.3 The Claims Grammar

After preprocessing the text may be finally parsed and translated by GF. Grammar that was written for that purpose was based on the Resource Grammar [2], but both restrictions and extensions were made. The restrictions refer to tenses and moods which would not be used in patent claims – as a preliminary analysis on our training corpus of 200 claims, where only verbs in present tense appeared.

The extensions are necessary because of the scientific nature of the text, which contains specific constructions, not found in the general-purpose resource grammar.

We will show the most common phenomena starting with some examples, first:

1. *a complement-fixing antibody* (also spelled without a hyphen) — 'an antibody that fixes a complement';
2. *a group comprising A, B and C* — 'a group that comprises A, B and C';
3. *purified rosette* — 'rosette that was purified by an undefined agent'.

Definitions in quotes show general pattern of translating such constructions, as they are not necessarily universal.

Grammar extended in that way was still insufficient. On one hand coverage was unsatisfactory, on the other hand, the huge number of ambiguities becomes a major problem.

GF is usually very fast parsing unambiguous texts and slow dealing with ambiguities. Let us observe, for example, the group *complement fixing antibody*, which can be parsed in two ways, either with *complement* or *antibody* as a head. This ambiguity will make a difference in translation, while many other ambiguities wouldn't.

Let us consider the phrase *human peripheral cells in the group comprising* ... Although this is completely equivocal there are a couple of possible parsing trees and two of them are visible on the figure 1. A careful reader shall easily find one more tree.

Large claims become unacceptably ambiguous, as the number of the interpretations is the cartesian product of the number of parse trees for each sub-constructions. An initial idea to reduce ambiguities is to introduce a richer syntactic hierarchy. For example observe that ambiguities in the sentence from the figure 1 could be avoided if AdjCN and QualifierPrepNP returned different types. Hence by adding new categories (Simple CN, CN with adjectives, CN with qualifiers) one may remove ambiguities and obtain fast and effective parsing. Unfortunatelly such a hierarchy proved impractical by forcing authors of the grammar to make a lot of arbitrary decisions (e.g. which category should one assign to *complement* in *complement-fixing antibody*). Finally adding a circular inclusion (CN with qualifiers into simple CN) was find inevitable.

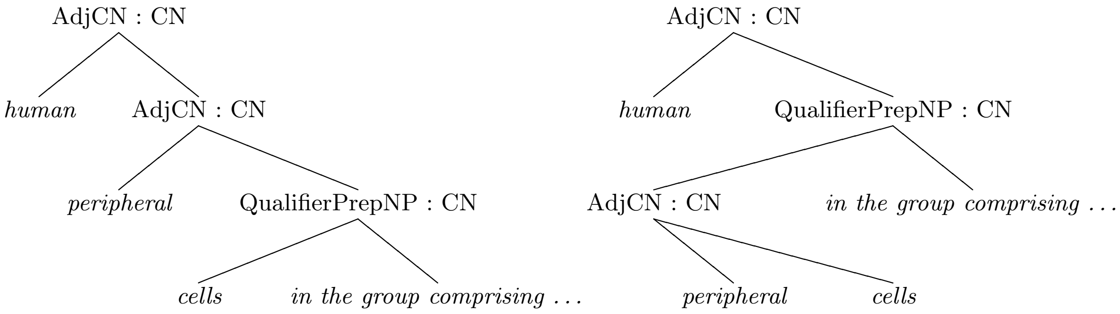

Fig. 1. Diagrams for *human peripheral cells in the group comprising ...*

Apart from ambiguities, small coverage is also a problem. Currently the coverage is about 15*compoundsgrammarandabout*35

Extensions of the Grammar Since the both the grammar coverage and the parsing efficiency are not satisfactory for scaling up the translation system, we decided to add robustness to the grammar by chunking patent claims into parseable pieces.

Our first approach uses the chunker that the GENIA tagger provides. Let us look at an example claim and how it is divided by GENIA[1]:

> Mouse complement-fixing monoclonal antibody which reacts with essentially all normal human peripheral T-cells but does not react with any of the normal human peripheral cells in the group comprising B cells , null cells and macrophages.

Chunker produces following phrases:

1. mouse complement-fixing monoclonal antibody,
2. reacts with essentially all normal human peripheral T-cells,
3. does not react with any of the normal human peripheral cells,
4. in the group comprising B cells,
5. null cells,
6. macrophages.

For the second chunk we obtain the following French translation: (2):

> réagisse avec essentiellement toutes les cellules AA périphériques humaines

However, because we need to have more controll over the possible interpretations of a claim, we should limit the usage of the chunker to situations where

[1] As a matter of fact GENIA chunks are smaller and presented phrases are results of some simple merging in postprocessing of the GENIA output

robustness would be too costly to obtain otherwise - such as splitting claims into sentences or identifying nested comments.

In this way, we can still keep all interpretations of a compound noun phrase, and use a statistical disambiguation tools or some set of probabilities to choose the right interpretation without ruling out additional possibilities.

4 Future Work

The work is still in progress and there are still many ways of making it scale up better that are still under construction. One of them is the chunker, as it is better to build one especially for this purpose, since we have access to the claims' structure, and moreover, we need to create a similar chunker for French also, and the two should have a consistently similar behaviour.

Also, the grammar needs a more thorough evaluation, in order to decide upon future extensions that would improve its coverage.

The disambiguation is an important step which is not currently implemented. An approach which is feasible given the current state of the GF runtime system is to rank the trees based on probabilities given to each syntactic rule. However, one needs a large treebank in order to get an approximation of the probabilities, which requires some manual work and also a more expressive grammar. Another approach is to learn probabilities based on their effect in translation. That is, for a given claim, to make the intersection between the parse trees of the English translation with the parse trees from the French translation. However, we need to experiment each of these methods in order to get an intuition of which would be a better heuristics to solve the problem.

References

1. Ranta, A.: Grammatical framework, a type-theoretical grammar formalism. Journal of Functional Programming **14**(2) (2004) 145–189
2. Ranta, A.: The gf resource grammar library. Linguistic Issues in Language Technology **2**(1) (2009)
3. Ceausu, A., Tinsley, J., Way, A., Zhang, J., Sheridan, P.: Experiments on domain adaptation for patent machine translation in the pluto project. Proceedings of the 15th Annual Conference of the European Association for Machine Translation (EAMT 2011) (2011)
4. Ehara, T.: Rule based machine translation combined with statistical post editor for japanese to english patent translation. MT Summit XI Workshop on patent translation, 11 September 2007, Copenhagen, Denmark (2007) 13–18
5. Ehara, T.: Statistical post-editing of a rule-based machine translation system. Proceedings of NTCIR-8 Workshop Meeting, June 15–18, 2010 (2010) 217—-220
6. Sheremetyeva, S.: On extracting multiword np terminology for mt. EAMT-2009: Proceedings of the 13th Annual Conference of the European Association for Machine Translation, ed. Lluís Màrquez and Harold Somers, 14-15 May 2009, Universitat Politècnica de Catalunya, Barcelona, Spain (2009) 205–212

7. Sheremetyeva, S.: “less, easier and quicker” in language acquisition for patent mt. MT Summit X, Phuket, Thailand, September 16, 2005, Proceedings of Workshop on Patent Translation (2005) 35–42
8. Sheremetyeva, S.: Natural language analysis of patent claims. In: Proceedings of the ACL-2003 workshop on Patent corpus processing - Volume 20. PATENT ’03, Stroudsburg, PA, USA, Association for Computational Linguistics (2003) 66–73
9. Sheremetyeva, S.: Handling low translatability in machine translation of long sentences. EAMT-2006: 11th Annual Conference of the European Association for Machine Translation, June 19-20, 2006, Oslo, Norway. Proceedings (2006) 105–114
10. Sheremetyeva, S.: Embedding mt for generating patent claims in english from a multilingual interface. MT Summit X, Phuket, Thailand, September 16, 2005, Proceedings of Workshop on Patent Translation (2005) 8–15
11. Tinsley, J., A.W., Sheridan, P.: Pluto: Mt for online patent translation. Proceedings of the 9th Conferences of the Association for Machine Translation in the Americas. Denver, CO, USA, 2010 (2010)
12. Tsuruoka, Y., Tateishi, Y., Kim, J.D., Ohta, T., McNaught, J., Ananiadou, S., Tsujii, J.: Developing a Robust Part-of-Speech Tagger for Biomedical Text. In Bozanis, P., Houstis, E.N., eds.: Advances in Informatics. Volume 3746. Springer Berlin Heidelberg, Berlin, Heidelberg (2005) 382–392
13. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of HLT-NAACL 2003. (2003) 252–259
14. Och, F.: Statistical machine translation live (04 2006)
15. Romary, L., Salmon-Alt, S., Francopoulo, G.: Standards going concrete: from lmf to morphalou. In: Proceedings of the Workshop on Enhancing and Using Electronic Dictionaries. ElectricDict ’04, Stroudsburg, PA, USA, Association for Computational Linguistics (2004) 22–28
16. Wikipedia: List of biomolecules — wikipedia, the free encyclopedia (2011) [Online; accessed 14-June-2011].