Emphasizing Syntax for French to German Machine Translation

Geoffray Bonnin, Violaine Prince

Emphasizing Syntax for French to German Machine Translation

Geoffray Bonnin[1, 2] Violaine Prince [2]
(1) Institut Wilhelm Schickard, Tübingen GERMANY
(2) University Montpellier 2 and LIRMM-CNRS 161 Ada Street 34392 Montpellier FRANCE
geoffray.bonnin@googlemail.com, prince@lirmm.fr

Abstract

This paper tackles the issue of automated translation from French to German, using syntactic analysis to enhance the results of lexical statistics approaches of these last years. It is a non symmetrical method aiming at producing correctly built sentences in the target language, from parsed sentences in source language. The idea is that translation between weakly divergent languages could be optimized relying on parsing output of the sentence to be translated. Then, after applying lexical transfer, the generation of the sentence in target language is performed through a set of light and recurrent transformations, applicable on the parsing tree structure. This paper describes a first study of divergence with an experiment on a corpus of aligned sentences extracted from the *Little Prince* of Saint-Exupéry.

1 Introduction

Since 1988, when the results of a purely statistic model for automated translation by IBM were published (Brown et al. 1990), computer based translation has been mostly relying on statistical approaches with two major tools : Multi-lingual lexical resources and aligned corpora. If lexical transfer from one language to another has strongly progressed since (Levin and Nirenburg 1994), translating full sentences, as complex syntactic and semantic units, still requires efforts. (Meyers et al. 2000) have shown that parsing source language sentences should enhance the quality of the target language output. This track has been followed by (Wu 1997), and then by (Yamada and Knight 01), with some success. But this approach still remains marginal, for two main reasons : (1) The majority of experiments tend to be focused on the lexical transfer which is in itself a heavy task, in spite of the advances carried out ((Simard et al. 2005)) ; (2) Syntactic analysis needs a robust parsing of the source language that goes beyond part-of-speech (POS) tagging, and this type of resource is not easily available.

In all cases, automated translation is a complex process, in which lexical, syntactic and semantic transfers are three major tasks to investigate. In this paper, we focus on the syntactic transfer task. The model presented here is inspired from (Prince and Chauché 2006). It points out the extra information provided by dependencies retrieval in a translation process. Its main features are sketched in next section which presents the SYGFtoE prototype thas has experimented translation between English (target) and French (source). Its most striking feature is that the parsing effort in the source language could be compensated by a light effort in generating the target language, provided that the considered pair of languages weakly diverges. In this paper, we describe an application considering the pair German (target) and French (source) in section 2. Since French and German diverge more than French and English, we will discuss the transformation effort necessary to provide a correct translation in section 3. Experiments have been run on a literary corpus, the *Little Prince* of Antoine de Saint-Exupéry, existing in both German and French. This choice has been made because this text is : (1) Completely and correctly parsed in French (see section 1), (2) correctly written in both French and German (we are not burdened with human stylistic or grammatical errors), (3) interesting as a set of different possible sentences constructions since it contains narrative parts, dialogues and thus, representative of some difficulties in sentences translation. (4) Represented in the form of an aligned corpus. Experiments results are shown and discussed in section 4.

2 SYGFtoE and the Divergence Theory

SYGFtoE is a prototype based on a relatively old translation architecture but which, with the current technological advances (processors speed, access to many resources, etc.), can be revisited with some success (Boitet 1999). It uses a syntactic analysis that provides POS tags and detects dependencies (complements, adverbials, subjects, objects) of each sentence in the source language, and builts the syntactic tree of the source sentence. The syntactic transfer process is based on a "light" generation in the target language, primarily using transformation operations as local as possible. The principle is to transform the parser output tree in such a way that the final tree state corresponds to the target language syntax. Once the structure is transformed, the real lexical transfer can be finished, because it is often dependent on grammar. The morpho-syntactic parser for French, SYGFRAN, has been developed by (Chauché 1984) and uses the SYGMART transformation engine (of the same author), a formal and recursive rewriting system based on Markov's algorithms applied to tree structures. It acts as a tree transducer, transforming any system with a complete formal description into another, by means of a transformation set of recursive grammars. SYGFRAN contains 12000 transformation rules representing French grammatical principles, and their application on textual entries transforms them into linguistic analysis trees. It has today an accuracy of 34% on any sentence. The remaining 66% divide in the following way : (1) Partially analyzed sentences (thus with completely specified tree structure portions and others under-specified). (2) Completely analyzed sentences but whose attachments are incorrect. There is no case in which SYGFRAN does not return any result. It participated to the EASY evaluation campaign (Paroubek et al. 2005) (syntactic parsers evaluation for French), begun in 2005, whose final results have been advertised in 2007 and where SYGFRAN has obtained a very good score. A good part of our work consisted in studying how to take advantage of the syntactic functions availability to model a better translation at the grammatical level.

2.1 Divergence theory

The SYGFoE model prototype assumes that **the translation effort from a language L1 to a language L2 is dependent on divergence between these two languages**. Divergence is by definition lexical, but it is also grammatical and stylistic. If the divergence theory had a good success at the end of the 80s and 90s ((Arnold 1993), (Levin and Nirenburg 1993) but especially (Dorr 1994)), it is currently abandoned mostly because, at that time, authors tried to solve the divergence problem by using models inspired from AI, to which an absence of quantitative results has been reproached by researchers favorable to the statistical approach. One part of our work consisted in defining a conceptual **structure divergence** starting from the transformation operations of tree structures, in the spirit of (Meyers et al. 2000). So, we operationally define syntactic divergence as follows. *The divergence between two tree structures A_1 and A_2 representing the trees of the same sentence respectively in the languages L_1 and L_2 is the number of transformation operations necessary to pass from A_1 to A_2*. The higher this number is, the broader the divergence. But it is not the only criterion : Indeed, it is important to know the scope of the structure divergence,i.e., the size of the portion of tree structure on which modifications appear.

2.1.1 Trees Transformation operations

The possible syntactic transformations are *insertion*, *inversion* or permutation, and *deletion*. They are also present in the prototype described in section 3. To illustrate them we show the transformation rules using non terminal symbols of the form `X(Y,[]) → T,U`. In this example, the tree substructure of root X and having for child at least the node Y is transformed entirely into a flat list including the two nodes T and U.

Insertion : insertion of a node into a tree substructure. The rule has the following form : `X(Y,[]) → X(Y,Z,[])`. The node Z may be inserted to the right or to the left of Y (the order will be respected). Here Z is inserted between Y and its brother nodes to the left. Example :

`GN(DETERM,JOURSEM) → GNPREP(an,DETERM,JOURSEM)`, e.g. *Le lundi → Am Montag* (*am* is the contraction of *an dem*), meaning "Monday"

Here, the preposition *an* has been inserted (the `GN` has thus been substituted by `GNPREP`).
Inversion or permutation : change of the order of the nodes in a tree substructure. These nodes may be of the same level or not. The rule has the following form :
`X(Y,W(Z,T))` → `X(Y,W(T,Z))`. (Permutation without change of level).
But it is also possible to have : `X(Y,W(Z,T))` → `X(T,W(Y,Z))`. (Permutation with change of level).
If one permutes two nodes of which one is root of a tree substructure, then all the children of these nodes are transported with it. Thus `X(Y,W)` → `X(W,Y)`, will put the children Z and T of W to the left of Y. Example :
`GN(N,GA)` → `GN(GA,N)` e.g. *une règle compliquée* → *eine komplizierte Regel*, meaning "a complicated rule".
Deletion : deletion of a node in a tree substructure. If this node is root in the original tree substructure, all its children are deleted too. The rule has the following form : `X([],Y,[])` → `X([],[])`. Example :
`GNPREP(mill,of,coffee)` → `GN(coffeemill)`
Notice that the example (from the English version) with terminal symbols as children (the lexical level provided by parsing and word to word transformation to the target lexicon) realises a double deletion.
Compared to SYGFtoE, this work has added another operation : **decoration**, which modifies nodes values. The above example carries out both a deletion (two nodes are removed) and a decoration : The prepositional noun phrase ($PREPNP$) becomes a simple noun phrase (NP). The same phenomenon exists in German. For example, `GN(der,20.,September)` (September 20th) becomes `PREPNP(an,der,20.,September)` in the sentence "Il y a eu trois meurtres le 20 septembre à Cologne" (three murders happened on September 20th in Koln). Here, decoration is done together with an insertion. There are cases in which decoration is not linked to another operation. It has been shown in (Bonnay 2006) that B. Dorr's divergence can be represented by combinations of transformation rules on syntactic trees generated by SYGFRAN.

2.1.2 Divergence Scope

An important aspect of SYGFtoE model also emphasized here, is the *divergence scope*. Rules can relate to subtrees of any size. A classification of the scope in three levels is thus proposed :
Constituent scope : rules implicating operations between nodes within a constituent (e.g. within a noun or verbal phrase)
Dependency scope : rules implicating operations between constituents
Maximal scope : Total divergence, mainly idiomatic expressions like :
Va te faire cuire un œuf → *Geh hin, wo der Pfeffer wächst*
(the first one literally means *go and cook yourself an egg* and the second *go where the pepper grows*).
If the constituent and dependencies divergence scopes can be grammatical (including morphological aspects) or syntactic, total divergence has a considerable stylistic range. In addition to the possible misinterpretation or the strangeness that can result from a word to word translation, the form of the target sentence must be well in the idiomatism of the target language.

2.2 SYGFtoE Functioning Principles

The translation of a sentence is processed according to the following steps :
(1)Syntactic analysis of the sentence producing a single tree
(2)Lexical transfer of the tree leaves : This step is not specified here because it is a complex task (a a first approach has been described by (Fessard 2006) , and we only focus here on the syntactic transfer task.
(3) Beginning of the Syntactic Transfer Task :Application of the constituent and dependency scope rules
(4)When necessary, application of maximal scope rules (translation of fixed metaphors or semi-fixed metaphors). These items take place one after the other willingly. Indeed, some metaphors can be partially modulated. For example : "Il a le bras long" (word to word "he has a long arm", metaphorically meaning "he is influent") can become "Il a le bras très long" (adding an adverb "very") without removing the metahoric sense. What must be translated in English by "He is influent" can thus be reinforced into "He is very influent" (what

SYGFtoE does). An purely dictionary-based approach may not recognize the expression.
(5)Generation of the sentence in English : in this step, the final sentence is built by going though the tree leaves, conjugating and declining their lemmas.

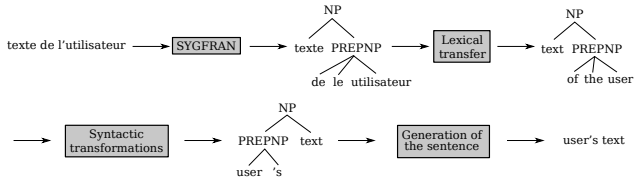

FIG. 1 – SYGFtoE's architecture

3 Designing the French to German Tree Transformation

The aim of this project was to study the feasibility of adapting SYGFtoE model to translation from French to German. We will show that divergence between French and German seems more marked than between French and English. In this paper we focus on the syntactic transformation part. A first step of lexical transfer has been dealt with by (Bonnay 2006).

3.1 Syntactic Parsing of Source Language

The output trees are obtained by sending requests to the online parser SYGFRAN. Its use is preferable because SYGFRAN is in perpetual evolution.The parser returns the syntactic tree in the following parenthesised form :

```
ELEM(
    VARLANFR,
    STR([1](tree structure)),
    VTQ(labels values),
    NOM_ETIQUETTES()
)
```

The preceding elements are defined as follows :
`VARANLFR` : name of the label defintion
`STR` : structures definitions (there can be up to 16 in the system, but for SYGFRAN only one is chosen(`[1]`)),
`VTQ` : labels values (variables with assigned values, defined in the `VARANLFR`lexicon to be read online),
`NOM_ETIQUETTES` : definition of the named labels (always empty for SYGFRAN).

3.2 Syntactic Transformation

3.2.1 Finding and writing the rules

When aligned corpora exist, the general idea is to see how, in the long term, semi-automatic techniques of syntactic transformation rules extraction could be set up. Nevertheless, to test SYGFtoE model on another pair of languages, a first set of syntactic transformations rules (which would be used to initiate a future training if the results were conclusive) was semi-automatically extracted from an aligned corpus out of the French and German versions of the Antoine de Saint-Exupéry's*Little Prince* . The syntactic transformation rules are assumed to be relatively few : A first empirical evaluation gives, for the number of transformation rules, a very small percentage of the number of rules necessary for parsing. This percentage is one of the possible measures of the grammatical and syntactic divergence modeling effort. If it exceeds a given value, then it can be more economical to use another method, like a pivot language, but if not then the method might be competitive. In the case of the French-English couple, SYGFtoE has given the following results : around 30 transformation rules were used for translating a corpus of 700 sentences taken randomly on the Internet, with a precision of 50%, where 400 parsing rules where used, giving a ratio of $7,5\%$((Prince and Chauché 2006)) for transformation effort, which is small (experiments are still running on several corpora to see if it stabilizes). The basic strategy to extract rules for the German-French pair was the following : Given two aligned sentences of the corpus, if the translation is not word to word, determine the syntactic transformations necessary. The rules which were written were verified in (Pittner and Berman 2004). Each rule consists of a tree and its transformation : $t_1 \rightarrow t_2$. The syntax of these trees in the implementation is the following : `1[`*node's attributes*`](`*children of* `1)` Examples :

`1[LEMMA(NP)](2[CAT(N)],3[LEMMA(AP)])` → `1(3,2)`
`1[LEMMA(PREPNP)](2[LEMMA(entfernt)],*)` → `1(*,2)`
`1[LEMMA(VP)](2[LEMMA(NP)],3[CAT(V)])` → `1(3,2)`

For more legibility, the rules will have a simplified writing in the rest of this article, when it is possible, i.e. when the nodes of the rules have one attribute which is not redundant. Thus, the first rule will be written : `NP(N,A)` → `NP(AP,N)`. It is sufficient to only consider the part to be transformed, which means that a node can have more children than those appearing in the left part of a rule. The latter

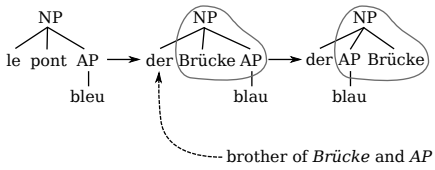

FIG. 2 – Successive transformations for the noun phrase"Le pont bleu" (the blue bridge)

will be indicated as being the **brothers** of the children appearing in the left part of a rule. For example, the rule above applies to `NP(der,Brücke,AP(blau))` (result of the syntactic analysis and the lexical transfer of *le pont bleu*, *the blue bridge* in English) and gives : `NP(der,AP(blau),Brücke)`. The determinant *der* is then a brother of *Brücke* and *AP* in the syntactic tree of the sentence to be translated (figure 2).

The "∗" and "X" symbols :

The "∗" symbol functions as follows : Given a tree node corresponding to the left part of a syntactic transformation rule, if one of its children is the "∗" symbol, any number of children of this node that are compared to it are of the same form as this child. For example the trees `a(b,c,d,e,f)` and `a(e,f)` are of the same form as the tree `a(*,e,f)`. The "`X`" symbol means that any tree is of the same form as it. The tree `a(b,c)` is of the same form as `a(X,c)`. The operations on trees described in the preceding section were largely used. To translate from French to German, complex operations were used like permutation with level change, and the triggering conditions can link terminal and nonterminal nodes. It is the case of the *dass* rule which imposes a rejection of the verb in last position : `CONJS(dass,NP,VP(V),*)` → `CONJS(dass,NP,VP,*,V)` Transformation rules can context dependent. They are also in a situation in which their application order can be important.

3.2.2 Rules priority

Some rules, like the *dass* rule, place a word in last position. However it happens that several rules of this type must be applied to the same sentence. It is thus necessary to take into account their priority. For example : *Je savais bien qu'il ne fallait pas l'interroger* (*I knew well that he should not be questioned*) must give in German *Ich wußte gut, dass man ihn nicht fragen dürfte*. This sentence implies placing *nicht*, the infinitive *fragen* and the modal *dürfte* in last position, because of the conjunction *dass*. The *dass* rule has thus priority on that of the modal (placement of infinitive in last position) which has itself priority on that of *nicht*. The solution is to write the rules in opposite order of their priority :

`CONJS(VP(ADVP(nicht),*)` →
`CONJS(VP,*,ADVP(nicht))`
`PCONJS(VP(MODAL,INFS(VP(V))),*)` →
`CONJS(VP(MODAL,INFS(VP)),*,V)`
`CONJS(DASS,NP,VP(V),*)` →`CONJS(DASS,NP,VP,*,V)`

which gives :

→ Ich wußte gut, dass man dürfte nicht ihn fragen.
→ Ich wußte gut, dass man dürfte ihn fragen **nicht**.
→ Ich wußte gut, dass man dürfte ihn nicht **fragen**.
→ Ich wußte gut, dass man ihn nicht fragen **dürfte**.

Designing and applying priorities is simplified because of the SYGMART engine structure that processes all these prototypes : Rules are grouped in ordered sets, called grammars, and the latter are applied by the engine according to their rank.

4 Experiments and Results

The whole parsed corpus contained 15,508 words in French, composing 1700 sentences. It was divided into a training and and test corpus of equal sizes. Once the model implemented and the rules extracted of the training corpus (on the basis of aligned sentences) it was tested on the second corups.The goal was to observe the regularity of rules and to modify them or to add another when necessary. The extracted rules scope tend to be a dependency scope(63%). The training corpus did not contain any total divergence. This result highlights the capacity of the model to measure the translation effort : German diverges relatively from French on the syntactic level and has thus many dependency scope rules. In the test corpus, only 30% was not correctly translated and needed increasing the extracted rules number by 23%. 18% of the extracted rules have been frequently reused (for several sentences) denoting thus a recurring transformation pattern. On the whole (with both corpora), around 20% of the sentences were word to word translations, which confirms the need to use syntactic analysis (80% used transformation rules). The results are summarized in the table.

On the whole (both corpora)	
Number of word to word translations	20%

First corpus (rules extraction)	
Constituent scope rules	37%
Dependency scope rules	63%
Maximum scope rules	0%

Second corpus (Test)	
Reused rules	18%
Number of sentences incorrectly translated by the extracted rules	30%
Number of added rules	23%

Experiments results for the syntactic transformation task show that on the given corpus, the divergence scope of the extracted rules is rather high (dependency scope). The rules reusability seems, for the moment, relatively small, but this is can be explained by the stylistic versatility of the corpus. The correct translations (on syntactic grounds) are rather good (70%). The effort to reduce errors is nonetheless important (23%), which confirms trends shown in most similar works : The last scores in percentage are the most difficult to achieve, because it corresponds to sophisticated structures or complex sentences.

5 Conclusion

This paper has only described the syntactic transformation part, where we had to built up a structure from scratch (we hope to increase the proportion of reusable rules through other experiments). Lexical transfer is currently dealt with and presents several difficulties. However, since many statistical approaches have brought up interesting results, and bilinguial lexical resources exist, we hope to achieve the task with less conceptual effort. Unfortunately we have no room here to detail it. Modeling syntaxtic transformation does not claim to solve the translation problem but it points out the importance of both the sentence (vs lexical) granularity, and the syntactic (vs semantic) dimension in translation, when considering quality. It could be used as a measure for the translation effort. Two ratios can be considered : One is the quantity of transformations compared to the parsing effort, and the other, the quantity of added transformations between two translated corpora. Both measures need to be studied with several experiments, and the best hope is that for a pair of languages they tend to respectively stabilize and become negligible.

References

C. Boitet. A research perspective on how to democratize machine translation and translation aids aiming at high quality final output. *Proceedings of MT Summit VII*. Pp 125-133. 1999

P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Laffety, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, pp 79–85, 1990.

G. Bonnin. *contribution à la traduction automatique franco-allemande*. mémoire de master recherche de l'Université Montpellier 2, 2006.

J. Chauché. Un outil multidimensionel de l'analyse du discours. *Proceedings of COLING84*, 1984.

D.Arnold. Sur la conception du transfert. *In P. Bouillon and A. Clas, editors, Agent, Etudes et Recherches en Traductique : Problemes de Traduction par Ordinateur*, 1993.

B. J. Dorr. Machine translation divergences : A formal description and proposed solution. *Computional Linguistics*, pp 597–633, 1994.

S. Fessard. *Transfert Lexical Français-Anglais*. mémoire de master recherche de l'Université Montpellier 2, 2006.

B. Levin and S. Nirenburg. Principles and idiosyncracies in mt lexicons. . *Notes of AAAI-93 Spring Symposium Series : Building Lexicons for MT*, 1993.

B. Levin and S. Nirenburg. The correct place of lexical semantics in interlingual machine translation. *Proceedings of COLING-94*, 1994.

A. Meyers, M. Kosaka, and R. Grishman. Chart-based transfer rule application in machine translation. *Proceedings of COLING2000*, pp 537–543, 2000.

K. Pittner and J. Berman. *Deutsche Syntax : ein Arbeitsbuch*. Narr Dr. Gunter, 2004.

V. Prince and J. Chauché. Translating through divergence : A application to french to english automatic translation. *RR LIRMM n.12758*, 2006.

P. Paroubek, L.G. Pouillot, I. Robba, and A. Vilnat. Easy : campagne d'évaluation des analyseurs syntaxiques. *Proceedings of TALN 05*, 2005.

M. Simard, N. Cancedda, B. Cavestro, M. Dymetman, E. Gaussier, C. Goutte, P. Langlais, A. Mauser, and K. Yamada. Traduction automatique statistique avec des segments discontinus. *Proceedings of TALN 05*, 2005.

D. Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, pp 377–403, 1997.

K.Yamada and K. Knight. A syntax based statistical translation model. *Proceedings of ACL-01*, 2001.