The adaptation of a machine-learned sentence realization system to French

Martine Smets, Michael Gamon, Simon Corston-Oliver and Eric Ringger
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
{martines, mgamon, simonco, ringger}@microsoft.com

Abstract

We describe the adaptation to French of a machine-learned sentence realization system called Amalgam that was originally developed to be as language independent as possible and was first implemented for German. We discuss the development of the French implementation with particular attention to the degree to which the original system could be re-used, and we present the results of a human evaluation of the quality of sentence realization using the new French system.

Introduction

Recently, statistical and machine-learned approaches have been applied to the sentence realization phase of natural language generation. The Nitrogen system, for example, uses a word bigram language model to score and rank a large set of alternative sentence realizations (Langkilde and Knight, 1998a, 1998b). Other recent approaches use syntactic representations. FERGUS (Bangalore and Rambow, 2000), Halogen (Langkilde 2000, Langkilde-Geary 2002) and Amalgam (Corston-Oliver et al., 2002) use syntactic trees as an intermediate representation to determine the optimal string output.

The Amalgam system discussed here is a sentence realization system which maps a semantic representation to a surface syntactic tree via intermediate syntactic representations. The mappings are performed with linguistic operations, the context for which is primarily machine-learned. The resulting syntactic tree contains all the necessary information on its leaf nodes from which a surface string can be read.

The promise of machine-learned approaches to sentence realization is that they can easily be adapted to new domains and ideally to new languages merely by retraining. The architecture of Amalgam was intended to be language-independent, although the system has previously only been applied to German sentence realization. Adapting this system to French allows us to assess which aspects of the system are truly language-independent and what must be added in order to account for French.

The purpose of this paper is to focus on the adaptation of Amalgam to French. Discussions about the general architecture of the system can be found in Corston-Oliver et al. (2002) and Gamon et al. (2002b).

1 Overview of German Amalgam

Amalgam takes as its input a logical form graph, i.e., a sentence-level dependency graph with fixed lexical choices for content words. This graph represents the predicate-argument structure of a sentence and includes semantic information concerning relations between nodes of the graph (Heidorn, 2002). Examples of French logical forms are given in section 3. Amalgam first degraphs the logical form into a tree and then augments it by the insertion of function words,

assignment of case and verb position features, syntactic labels, etc., to produce an unordered syntax tree. Amalgam then establishes intra-constituent order. After syntactic aggregation, insertion of punctuation, morphological inflection, and capitalization, an output string is read off the leaf nodes. The contexts for most of these linguistic operations are machine-learned (Gamon et al., 2002a). Figure 1 lists the eight stages in German Amalgam: the label *ML* denotes that the operation is applied in machine-learned contexts, and the label *Proc* indicates that the operation is procedural or deterministic.

Stage 1 Pre-processing (Proc)
- degraphing of the semantic representation
- retrieval of lexical information

Stage 2 Flesh-Out (ML):
- assignment of syntactic labels
- insertion of function words
- assignment of case and verb position features

Stage 3 Conversion to syntax tree (Proc):
- introduction of syntactic representation for coordination
- splitting of separable prefix verbs based on both lexical information and previously assigned verb position features

Stage 4 Movement:
- raising, wh movement (Proc)

Stage 5 Ordering (ML):
- ordering of constituents and leaf nodes in the tree

Stage 6 Extraposition (ML)

Stage7 Surface clean-up (ML):
- lexical choice of determiners and relative pronouns
- syntactic aggregation

Stage 8 Punctuation (ML)

Stage 9 Inflectional generation (Proc)

Figure 1 The stages of German Amalgam

All machine-learned components employ decision trees for classification and for probability distribution estimation (Gamon et al., 2002b). The decision trees are built with the WinMine toolkit (Chickering, 2002). There are a total of twenty-one decision trees in the German system. The complexity of the decision trees varies with the complexity of the modeled task: the number of branching nodes in the decision tree models in the German system ranges from just 4 to.7,876 in the order model.

2 Data and feature extraction

The data for all models are automatically extracted from of a set of 100,000 sentences drawn from software manuals. Between 30,000 and one million cases are extracted from these sentences, depending on the task to be modeled. The sentences are analyzed in the NLPWin system (Heidorn, 2002), which provides a syntactic and logical form analysis. Nodes in the logical form representation are linked to the corresponding syntax nodes, allowing us to learn contexts for the mapping from the semantic representation to the surface syntax representation. The data is split 70/30 for training versus model parameter tuning. For each set of data we build decision trees at several levels of granularity and select the model with the maximal accuracy as determined on the parameter tuning set.

We attempt to standardize as much as possible the set of features to be extracted. We exploit the full set of features and attributes available in the analysis, instead of pre-determining a small set of potentially relevant features for each model. This allows us to share the majority of code among the individual feature extraction tasks and among languages. Typically, we extract the full set of available linguistic features of the node under investigation, its parent and its grandparent, with the only restriction being that these features need to be available at the stage where the model is consulted at generation run-time. This yields approximately six hundred features that provide a sufficiently large structural context for the operations. In addition, for some of the models we add a small set of specially computed linguistic features that we believe to be important for the task at hand.

3 French Amalgam

French Amalgam re-uses the architecture of the German system. Indeed, sentence realization from a semantic graph must undergo many of the same transformations regardless of the language: pre-processing of the logical form, fleshing-out,

conversion to syntax tree, etc. We outline below the stages of the French system, and compare them to the German system.

Stage 1, the pre-processing of the data, involves language-neutral transformations from a graph representation to a tree representation, and can be reused without alteration by the French system.

The fleshing out of the logical form in Stage 2 required changes for French. French does not need a machine-learned model for case. On the other hand French requires a model for clitic insertion which does not exist in German. Language-specific details of feature selection for Stage 2 will be discussed in section 3.2.

Because French does not have separable prefix verbs, the lexical operation that splits prefixes in German is not needed in Stage 3. French uses a head-switching operation for verb phrases headed by modal verbs, because of the status of French modals as presented in section 3.1.3.

Stage 4 (raising and Wh movement) is identical for both languages.

In stage 5, both German and French use a left-to-right model of constituent order. For each language, the model is a decision tree representing the probability distributions involved in ordering (see Ringger et al. (in preparation) for a detailed discussion of different approaches to constituent ordering).

Extraposition, which is common in German (Gamon et al. 2002c), is rare in the French technical software manuals: there were too few examples of extraposition in the French data to train an extraposition model for Stage 6.

Stage 7 (clean-up) uses language-specific information, especially in the realization of lexical forms of function words.

Finally, stage 8, the realization of inflection, is completely language specific.

Figure 2 provides a summary of the French Amalgam system.

Stage 1 Pre-processing (Proc):
- Degraphing of the logical form.
- Retrieval of lexical information.

Stage 2 Flesh-Out (ML):
- Assignment of syntactic labels.
- Insertion of function words.
- Insertion of clitics.
- Assignment of case (Proc).

Stage 3 Conversion to syntax tree (Proc)
- Introduction of syntactic representation for coordination.
- Head-switching (ML).

Stage 4 Movement:
- Raising, wh movement (Procedural).

Stage 5 Ordering (ML):
- Ordering of constituents and leaf nodes in the tree.

Stage 6 Surface clean-up (ML):
- Lexical choice of determiners and relative pronouns.
- Syntactic aggregation.

Stage 7 Punctuation (ML)

Stage 8 Inflectional generation (Proc)

Figure 2 The stages of French Amalgam

There are eighteen decision trees in the French system, and the complexity of the decision trees varies with the complexity of the task modeled. The number of branching nodes in the decision tree models in the French system ranges from 6 to 838, except for the order model which has 4682 branching nodes.

There are a number of differences between the systems, some concerning models that are language-specific, others relating to features relevant only for one language. Most of the differences are in feature extraction and in the linguistic operations relying on the information provided by the models. We discuss these differences in the following sections

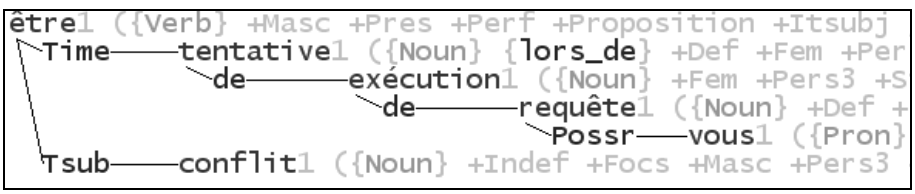

Figure 3 French logical form illustrating the *il y a* construction

3.1 Models

In this section we discuss the solutions adopted for the treatment of case, clitics and modals in the French Amalgam system.

3.1.1 Case

As mentioned above, French generation does not need a model for case, since case does not exist anymore in French, except for some traces in the pronominal system. Determining case for pronouns is a trivial task in French that does not require a machine-learned solution.

For example, *le* is the form of the third person singular object clitic (accusative), while *lui* is the third person singular indirect object (or dative). A knowledge-engineered module determines the case of each pronoun on the basis of the predicate-argument structure in our linguistic representation, whereas the German system uses a decision tree model to assign case to all nominal constituents.

3.1.2 Clitics

French clitics that function as arguments of the verb are represented directly in the logical form. Clitics that are used expletively are not represented in the logical form, and thus need to be inserted during sentence realization. For example the clitic *y* in *il y a* ("there is"), is inserted during the flesh-out stage. An example is given in (1), with the logical form in Figure 3.

(1) Il y a eu un conflit lors de la tentative d'exécution de votre requête
"There was a conflict at the time of execution of your request."

In the logical form in Figure 3, the verb *avoir* ('have') is represented by *être* ("be") and there is no clitic *y* ('there'), nor subject *il* ('it') . The clitic *y* thus needs to be inserted in that representation (as well as the expletive subject). The context for *y*-insertion is learned and represented in the clitic decision tree. This component is necessary for French and would be needed for other Romance languages also.

3.1.3 Modals

Most of the changes in the French system were required by modal verbs. In German and English syntactic analysis, modals do not head a clause but behave like auxiliaries. In French, however, modals behave syntactically just like main verbs (taking a clausal complement), but have semantic properties characteristic of modals: *pouvoir* ("to be able") and *devoir* ("to have to"), for example, can be used epistemically or deontically (Palmer, 1986). In our system, they share the same semantic representation as modals of other languages, although they do not exhibit the same syntactic behavior.

In example (2), the modal *pouvez* (a second person plural form of the verb *pouvoir*) is the head of the main clause, and carries the syntactic information of tense, mood, person number and negation. The logical form for this example, illustrated in **Figure 4**, is headed by *envoyer* ("to send"). The modal *pouvoir* ("to be able") functions in the logical form as a semantic modifier of the verb, and features such as tense, mood, and negation are copied onto the semantic head, *envoyer*.

(2) Vous ne pouvez pas envoyer un message à plusieurs personnes en même temps.
"You cannot send a message to several people at the same time.

Figure 4 French logical form illustrating a modal construction

During the generation process, the modal verb has to become a syntactic head with a clausal complement to reflect the syntactic properties of French. The contexts for the operation which switches semantic and syntactic headedness are learned automatically. When a switch is predicted, a knowledge-engineered module is invoked to perform the switch and make the necessary structural adjustments.[1]

3.2 Differences in features

Each decision tree must be trained for the given language. Consequently, the decision trees produced for each language may differ in their target feature values or may require language-dependent feature extraction.

The decision tree classifiers for the French system are trained on a corpus of 100,000 sentences, drawn from technical software manuals in French. The models are tested on a test corpus of the same domain (but distinct from the training corpus). All the examples in this paper come from that technical corpus.

3.2.1 Target feature values

Target feature values in many instances refer to specific lexical items and are therefore language specific. For example, for the insertion of constituents such as auxiliaries, prepositions, and infinitive markers, the value of the target feature is the citation form of the word being inserted. Thus, for most of the models of stage 2 (flesh-out), the definition of the target feature is language specific.

There are some cases, however, where the value of the target feature is language independent. The most obvious case is the syntactic labeling of constituents such as NP, PP, etc. Other examples include models with yes/no target feature values, such as the model which determines the probability that certain NPs are not syntactically realized (for examples, the subjects of infinitive or imperatives). In these cases, the code defining these target features can be re-used for a number of languages without change.

[1] The switch operation also applies to partitive constructions. Support verbs, on the other hand, have the same representation as other verbs in our logical form.

3.2.2 Feature extraction

As noted previously, German feature extraction modules have been re-used for the French system. Feature extraction was unchanged (albeit performed on French data) for the models that capture the contexts for the insertion of negation, prepositions, and subordinating conjunctions. However, in every one of these cases, the set of values for the target features changed to reflect the language.

Most models, however, require slightly different sets of extracted features. For example, the French model responsible for the realization of the determiner needs to check for the presence of an adjective in between the determiner and the noun. The form of the plural indefinite determiner is *de* before an adjective or *des* immediately before the noun. Besides that, the model refers only to the gender and number of the head noun. In German, however, the form of the determiner is determined by the gender, number and case of the head noun.

The model which determines the realization of the relative pronoun also looks at the gender of the pronoun's antecedent in French, because some pronouns agree in gender and number with their antecedent. For example, in (3), the feminine form *laquelle* ("which") agrees with its antecedent, *base de données*.

(3) Développez Bases de données, puis développez la base de données à laquelle appartient l'utilisateur.
"Expand Databases, then expand the database to which the user belongs."

However, case information is not useful to determine the form of the pronoun, even though subject and object relative pronouns are marked for case. *Qui* is the subject form and *que* the object form of the relative pronoun, but in many cases, *qui* is used to refer to a human antecedent with any syntactic function (*avec qui* "with whom", *pour qui* "for whom", etc.), and not to the subject of the clause. Hence, this distinction of forms, a vestige of erstwhile case marking, is not relevant to automatically distinguish uses of relative pronouns and is not amongst the extracted features. Grammatical function information is used instead by the decision tree learner.

To determine the insertion of expletive subjects, French specific information was necessary. The most common context of insertion is with *être* ("to be"), and a feature specific to that environment was added to the set of extracted features.

For determining the syntactic label of a constituent, more information again is needed in French, because of the nature of French modals. Verbs used with a modal must be marked as such, otherwise they are assigned the label of the head instead of the label of its complement. (Note that the assignment of syntactic labels takes place before the operation that switches semantic and syntactic headedness).

The examples above involve some features which are not relevant for German, or which are specific to French. In all cases, however, French uses the same strategy as German: exploit the full set of features available in the analysis on the node, its parent and grand-parent. The sets of features therefore largely overlap and are language-independent for the most part. About 700 features are extracted for most of the French models, 1000 for the order model. These sets include syntactic features (category, arguments, syntactic function, subcategorization features, etc.), morpho-syntactic features (agreement features, tense, mode, aspect features) and semantic features (semantic roles, semantic relations). A subset of features are selected as relevant during the learning of each decision tree classifier: complex models have over 100 features (120 and 177 features for the label model and the order model respectively), simpler models use much fewer features (12 for the clitic model, 13 for the relative pronoun realization model and 3 only for the switch model). The features selected in the relative pronoun model, for example, are the syntactic category of the node, of its parent, the syntactic function of the node, the voice of the parent, the arguments of the parent, and the agreement features of the grand-parent. These features correspond to linguistic intuition: the choice of a relative pronoun depends on its syntactic category and on the function it fulfills. Its agreement features depend, in French, on the agreement features of the constituent modified by the relative clause.

Details of some models are given in the appendix, with relevant statistics. The next section briefly discusses linguistic operations which rely on machine-learned contexts.

3.3 Linguistic operations

Most of the linguistic operations which are employed in mapping a semantic representation to a syntactic tree have machine-learned contexts. Once the operation is triggered in a given context, the action part of the operation contains language-specific elements, such as specific lexical choices for function word insertions, etc. While the structure of most of these operations could be re-used for the French implementation, some adaptations had to be made.

Linguistic operations which insert constituents are often very similar, and differ only, in some cases, in the citation form of the lexical element being inserted. For example, specific prepositions or infinitival markers are inserted. The definitions of these operations are thus very close in German and in French. This is not the case, however, for configurations where modals can occur, and which necessitate the definition of special cases for French modals. Also, although the conversion of the logical form to a syntactic tree is language independent for the most part, the operation which switches syntactic and semantic headedness involves many specifications for the contexts of French modals. The last stage of sentence realization, inflection, is also completely language-dependent.

The operations of the ordering stage and of the surface-cleanup stages, on the other hand, are completely language-neutral, albeit based on machine-learned models trained on French data.

4 Evaluation

We performed a human evaluation of French generation. This was the first formal evaluation of the French generation system. For this evaluation, 545 test sentences from a blind software manual corpus were analyzed with our NLPWin analysis system, producing a logical form for each sentence. From each logical form, our sentence realization system then generated a hypothesis sentence. We did not control for noise introduced into the data by the analysis phase (about 15% of the sentences did not have a spanning parse). Nevertheless, this experiment

gives us a good indication of the performance of French Amalgam.

Five evaluators were asked to evaluate the same set of sentences independently. Each generated sentence was evaluated in isolation; i.e., discourse context was not taken into account. For each sentence, raters were presented with the original French sentence as a reference and the hypothesis sentence from French Amalgam. All the raters assigned an integer score comparing each sentence to the reference. The scores were 1 "Unacceptable", 2 "Possibly acceptable", 3 "Acceptable" and 4 "Ideal".

The score of a sentence is the average of the scores from the five raters. The system score is the average of the scores of all sentences.

The average score was 2.92 with a standard deviation of 0.19. The maximum score was 4, and 99/545 sentences (18.2%) received that score. For 45 of those sentences, the score was assigned automatically, because the sentences were completely identical. The other sentences with score 4 (54 sentences in total) differed in some way from the original but had been assigned that score by all 5 evaluators, who had judged them equivalent to the reference sentence.

5 Conclusion

We have discussed the adaptation to French of a machine-learned sentence realization system, originally developed for German generation. We have shown that, thanks to the language-independent architecture and the machine-learning orientation of the system, we were able to re-use most of the original code. Feature extraction and model building are language-neutral, with the exception of the addition of French-specific features. All remaining differences are in the specific linguistic operations which map the semantic representation to the generated string and are limited to specific lexical choices or to reverting semantic and syntactic headedness in modal contexts. Of course, a few components are relevant only for one of the languages (such as the clitic model in French), but these are very few.

The results of the evaluation are very encouraging: they are comparable to the results for German sentence realization, reported in Corston-Oliver et al. (2002): 2.96, with a standard deviation of 0.81, with a similar rating system.

Finally, it should be noted that the total development time for adapting the system from German to French was ten person-weeks. This time includes training all of the models, and general improvements in the system.

6 Acknowledgements

Our thanks go to the five anonymous, independent evaluators for assistance with evaluation and to the Microsoft Research NLP group.

References

Bangalore S. and Rambow O. (2000) "Exploiting a probabilistic hierarchical model for generation". In *Proceedings of COLING 2000*, Saarbrücken, Germany, pp. 42-48.

Chickering D. M. (2002) *The WinMine Toolkit.* Microsoft Technical Report MSR-TR-2002-103.

Corston-Oliver S., Gamon M., Ringger E. and Moore R. (2002) "An overview of Amalgam: a machine-learned generation module". In *Proceedings of INLG 2002*, New York, pp.33-40.

Gamon M., Ringger E., Corston-Oliver S., Moore R. (2002a) "Machine-learned contexts for linguistic operations in German sentence realization". In *Proceedings of ACL 2002*, pp. 25-32.

Gamon M., Ringger E. and Corston-Oliver S. (2002b) *Amalgam: A machine-learned generation module.* Microsoft Research Technical Report MSR-TR-2002-57.

Gamon M., Ringger E., Zhang Z., Moore R. and S. Corston-Oliver (2002c) "Extraposition: A case study in German sentence realization". In: *Proceedings of COLING 2002*, pp. 301-307.

Heidorn G. E. (2002) "Intelligent Writing Assistance". In R. Dale, H. Moisl, and H. Somers (eds.), *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text*, Marce Dekker, New York.

Langkilde I. (2000) "Forest-Based Statistical Sentence generation". In *Proceedings of NAACL 2000*, pp. 170-177.

Langkilde-Geary I. (2002) "An Empirical Verification of Coverage and Correctness for a General-Purpose Sentence Generator". In *Proceedings of INLG 2002*, New York, pp.17-24.

Langkilde I. and Knight K. (1998a) "The practical value of n-grams in generation". In *Proceedings of the 9th International Workshop on Natural Language Generation*, Niagara-on-the-Lake, Canada pp. 248-255.

Langkilde I. and Knight K. (1998b) "Generation that exploits corpus-based statistical knowledge". In *Proceedings of the 36th ACL and 17th COLING*. Montreal, Canada pp. 704-710.

Palmer F. (1986) *Mood and Modality*, Cambridge University Press, Cambridge.

Ringger E., Gamon M., Smets M., Corston-Oliver S. and Moore R. (in preparation) "Linguistically informed models of constituent structure for ordering in sentence realization".

Appendix: Details on a subset of the decision tree models in French Amalgam

Model	**Values predicted**	**Accuracy**	**Baseline**
Syntactic label	Determiner phr., complement cl., VP, quantifier phr., adverbial NP, imperative main cl., adverb phr., label, appositive NP, question main cl., nominal relative, adjective phr., relative cl., NP, possessor, present participial cl., comment, infinitival cl., PP, finite subordinate cl., declarative main cl., past participial cl., present participial cl., absolute clauses.	0.9925	0.3087
Placeholder for determiner	NULL, Wh, proximal demonstrative, definite, indefinite	0.9892	0.6167
Auxiliary	NULL, être-avoir, avoir-mod, être, avoir	0.9979	0.9132
Prepositions	NULL, de, par	0.9965	0.9793
Insert infinitive marker	NULL, de, pour, à	0.9315	0.4024
Insert negator	NULL, ne, ne_pas	0.9057	0.7188
Realization of NP	Yes, No	0.8871	0.6625
Insert clitics	NULL, y, en	0.9981	0.9975
switch head	Yes, No	0.9971	0.6606
Determiner form	le, les, l', la, un, une, des, de, d', du, ce, cet, cette, ces, cettes, quel, quelle, quels, quelles	0.9894	0.2705
Relative pron. form	qui, où, dont, que, quoi, lequel, laquelle, lesquels, lesquelles	0.9326	0.5303
Conjunction reduction	Spell out: first or last instance	0.9557	0.6739
Order: move constituent	Yes, No	0.9798	0.6272