Weather Report Translation using a Translation Memory

Thomas Leplus, Philippe Langlais and Guy Lapalme

RALI/DIRO
Université de Montréal
C.P. 6128, succursale Centre-ville
H3C 3J7, Montréal, Québec, Canada
http://www.rali-iro.umontreal.ca

Abstract. We describe the use of a translation memory in the context of a reconstruction of a landmark application of machine translation, the Canadian English to French weather report translation system. This system, which has been in operation for more than 20 years, was developed using a classical symbolic approach. We describe our experiment in developing an alternative approach based on the analysis of hundreds of thousands of weather reports. We show that it is possible to obtain excellent translations using translation memory techniques and we analyze the kinds of translation errors that are induced by this approach.

1 Introduction

In the mid seventies, a group of linguists and computer scientists of Université de Montréal (*TAUM* group) developed an English to French weather report machine translation system which became known as TAUM-MÉTÉO, described in [8, chap12]. It involves three major steps: a dictionary look-up, a syntactic analysis and a light syntactic and morphological generation step.

The transfer from English to French is encoded at the word level in three special purpose lexicons: idioms (e.g. *blowing snow* ↔ *poudrerie*) , locations (e.g. *Newfoundland* ↔ *Terre Neuve*) and a general dictionary containing syntactic and semantic features (e.g. *amount*=N((F,MSR),*quantite*) which means that *amount* translates into the feminine F French noun N *quantite* which is a measure MSR noun).

The syntactic stage is the result of a detailed analysis that was done by hand at the early stage of the prototype. [3] reports that MÉTÉO-2, a subsequent system that became operational at Environment Canada, used fifteen different grammars categorised into five major types from which the syntactic analysis chooses the most appropriate one.

The third and last step performs French word reordering (e.g. adjectives are placed after the noun they modify), preposition selection (e.g. we say ***à*** *Montréal* but ***en*** *Nouvelle-Écosse* and ***au*** *Manitoba*) plus a few morphological adjustments (e.g. *le été* → *l'été*).

This system has been in continuous use since 1984, translating up to 45,000 words a day. [7] argues that one of the reasons for the success of the MÉTÉO system is the nature of the problem itself: a specific domain, with very repetitive texts that are particularly unappealing to translate for a human (see for example the reports shown in Figure 1). Furthermore, the life of a weather report is, by nature, very short (approximatively 6 hours), which makes them an ideal candidate for automation.

Professional translators are asked to correct the machine output when the input English text cannot be parsed, often because of spelling errors in the original English text. MÉTÉO is one of the very few machine translation systems in the world from which the unedited output is used by the public in everyday life without any human revision.

Some alternatives to machine translation (MT) have been proposed for weather reports, namely multilingual text generation directly from raw weather data: temperatures, winds, pressures etc. Such generation systems also need some human template selection for organising the report. Generating text in many languages from one source is quite appealing from a conceptual point of view and has been cited as one of the potential applications for natural language generation [15]; some systems have been developed [9, 6, 4] and tested in operational contexts. But thus far, none has been used in everyday production to the same level as the one attained by MT. One of the reasons for this is that meteorologists prefer to write their reports in natural language rather than selecting text structure templates.

Our goal in this study was to determine how well a simple memory-based approach would fit in the context of the weather report translation. We describe in section 2 the data we received from Environment Canada and what preprocessing we performed to obtain our MÉTÉO bitext. We present in section 4 the first prototype developed and then report on the results obtained in section 5. We analyse in section 6 the main kinds of errors that are produced by this approach. In section 7, we conclude with a general discussion of this study and propose some possible extensions.

2 The corpus

We obtained from Environment Canada forecast reports in both French and English produced during 2002 and 2003. The current reports are available on the web at `http://meteo.ec.gc.ca/forecast/textforecast_f.html`.

We used this corpus to populate a bitext i.e. an aligned corpus of corresponding sentences in French and English weather reports. Like all work on real data, this conceptually simple task proved to be more complicated than we had initially envisioned. This section describes the major steps of this stage.

We received files containing both French and English weather forecasts. Both the source report, usually in English, and its translation, produced either by a human or by the current MÉTÉO system, appear in the same file. One file contains all reports issued for a single day. A report is a fairly short text, on

```
FPCN18 CWUL 312130

SUMMARY FORECAST FOR WESTERN QUEBEC
ISSUED BY ENVIRONMENT CANADA

MONTREAL AT 4.30 PM EST MONDAY 31
DECEMBER 2001 FOR TUESDAY 01 JANUARY
2002. VARIABLE CLOUDINESS WITH
FLURRIES. HIGH NEAR MINUS 7.

END/LT
```

```
FPCN78 CWUL 312130

RESUME DES PREVISIONS POUR L'OUEST
DU QUEBEC EMISES PAR ENVIRONNEMENT
CANADA

MONTREAL 16H30 HNE LE LUNDI 31
DECEMBRE 2001 POUR MARDI LE 01
JANVIER 2002. CIEL VARIABLE AVEC
AVERSES DE NEIGE. MAX PRES DE MOINS
7.

FIN/TR
```

Fig. 1. An example of an English weather report and its French translation.

average 304 words, in a telegraphic style. All letters are capitalised and non accented and almost always without any punctuation except for a terminating period.

As can be seen in the example in Figure 1, there are few determiners such as articles (*a* or *the* in English, *le* or *un* in French). A report usually starts with a code identifying the source which issued the report. For example, in `FPCN18 CWUL 312130`, `312130` indicates that the report was produced at 21h30 on the 31st day of the month; `CWUL` is a code corresponding to Montreal and the western area of Quebec. A report (almost always) ends with a closing markup: `END` or `FIN` according on the language of the report. If the author or the translator is a human, his or her initials are added after a slash following the markup.

Our first step is to determine the beginning and end of each weather forecast using regular expressions to match the first line of a forecast, which identifies the source which issued it, and the last line which usually starts with `END` or `FIN`.

Then we distinguish the English forecasts from the French ones according to whether they ended with `END` or `FIN`. Given the fact that we started with a fairly large amount of data, we decided to discard any forecast that we could not identify with this process. We were left with 273 847 reports.

Next, we had to match English and French forecasts that are translations of each other. As we see in fig. 1, the first line of the two reports is almost the same except for the first part of the source identifier which is `FPCN18` in English and `FPCN78` in French. After studying the data, we determined that this shift of 60 between English and French forecast identifiers seemed valid for identifiers from `FPCN10` through `FPCN29`. These identifiers being the most frequent, we decided to keep only these into our final bitext.

This preprocessing stage required about 1 500 lines of Perl code and few weeks of monitoring. Of the 561 megabytes of text we originally received, we were left with *only* 439 megabytes of text, representing 89 697 weather report pairs.

To get a bitext out of this selected material, we first automatically segmented the reports into words and sentences using an in-house tool that we did not try to adapt to the specificity of the weather forecasts.

We then ran the Japa sentence aligner [11] on the corpus (this took around 2 hours running on a standard P4 workstation), to identify 4,4 million pairs of sentences, from which we removed about 28 000 (roughly 0.6%) which were not one-to-one sentence pairs.

We divided this bitext into three non-overlapping sections, as reported in Table 1: TRAIN (January 2002 to October 2003) for populating the translation memory, BLANC (December 2003) for tuning a few meta-parameters, and TEST (November 2003) for testing.

The TEST section was deliberately chosen so as to be different from the TRAIN period in order to recreate as much as possible the working environment of a system faced with the translation of new weather forecasts.

		English		French	
corpus	pairs	words	toks	words	toks
TRAIN	4 211	30 493	10.0	37 542	11.1
BLANC	122	888	3.0	1 092	3.2
TEST	36	269	1.9	333	2.0
total	4 370	31 383	10.1	38 968	11.3

Table 1. Main characteristics of the subcorpora used in this study in terms of number of pairs of sentences, English and French words and tokens. Figures are reported in thousands.

A quick inspection of the bitext reveals that sentences are fairly short: an average of 7.2 English and 8.9 French words. Most sentences are repeated: only 8.6% of the English sentences unique. About 90% of the sentences to be translated can be retrieved verbatim from the memory with at most one edit operation i.e. insertion, suppression and substitution of a word. These properties of our bitext naturally suggest a memory-based approach for translating weather reports.

3 The translation memory

Our translation memory is organised at the sentence level. We define a memory ($\mathcal{M}$) as a set of (M) triplets, where a source sentence (e_i) is associated to its N most frequent translations ($f_1^i, \ldots, f_N^i$), each translation f_j^i being associated with its count (n_j^i) of cooccurrence with e^i in the memory:

$$\mathcal{M} = \left\{ \left(e^i; f_1^i, \ldots, f_N^i; n_1^i, \ldots, n_N^i \right) \right\}_{i \in [1,M]}$$

Two parameters could significantly affect the performance of the memory: its size (M) and the number (N) of French translations kept for each English sentence.

The size of the translation memory affects our system in two ways. If we store only the few most frequent English sentences and their French translations, the time for the system to look for entries in the memory will be short. But, on the

other hand, it is clear that the larger the memory, the better our chances will be to find sentences we want to translate (or ones within a short edit distance), even if these sentences were not frequent in the training corpus.

The percentage of sentences to translate found verbatim in the memory grows logarithmically with the size of the memory until it reaches approximately 20 000 sentence pairs. With the full memory (about 300 000 source sentences), we obtain a peek of 87% of sentences found into the memory.

The second parameter of the translation memory is the number of French translations stored for each English sentence. Among the 488 792 different English sentences found in our training corpus, 437 418 (89.5%) always exhibit the same translation. This is probably because most of the data we received from Environment Canada is actually machine translated and has not been edited by human translators. In practice, we found that considering a maximum of $N = 5$ translations for a given English sentence was sufficient for our purposes.

4 The translation procedure

The overall scenario for translating a sentence is the following. The source sentence is first preprocessed in order to handle more easily entities such as dates, hours or numbers. Note that this process has been done as well for the sentences populating the memory. The source sentences which are closest to this processed source sentence are then retrieved from the translation memory. This leaves us with a set of target sentences from which we select the best ones . The chosen translation is then postprocessed to remove the meta-tokens introduced by the preprocessing phase. We now briefly discuss each of these steps.

Eight classes of tokens (*punctuation*, *telephone numbers*, *months*, *days*, *time*, *numeric values*, *range of values* and *cardinal coordinates*) are identified via regular expressions at the character level and replaced by a corresponding tag (numbered if there are more than one of the same class in a sentence). This process is quite different from the creation of specialised lexicons used in the current MÉTÉO system. In particular, we did not explicitly encode place names.

For a given sentence to translate e, a set of the 10 closest source sentences (in terms of edit distance) in the memory is first computed: $\mathcal{E} = \{\overline{e}_1, \ldots, \overline{e}_{10}\}$. Since each of these source sentences is associated with at most $N = 5$ translations, we retrieve from the memory a set $\mathcal{F}$ of at most 50 candidate translations and their associated count. Note that several target sentences might be identical in the set:

$$\mathcal{F} = \left\{ \langle (f_1^c, \ldots, f_N^c), (n_1^c, \ldots, n_N^c) \rangle_{c \in [1,10]} \right\}$$

The target sentences f_j^c are then ranked in increasing order of edit distance between their associated English sentence e^c and the source sentence e. Ties in this distribution are broken by prefering larger counts n_j^c.

The translations produced are finally postprocessed in order to transform the meta-tokens introduced during preprocessing into their appropriate word form. We observed on a held-out test corpus that this cascade of pre- and postprocessing clearly boosted the coverage of the memory.

5 Results

We report in Figure 2 the performance of the engine measured on the TEST corpus in terms of Word Error Rate (WER) and Sentence Error Rate (SER) as a function of the number (M) of pairs retained in the memory.

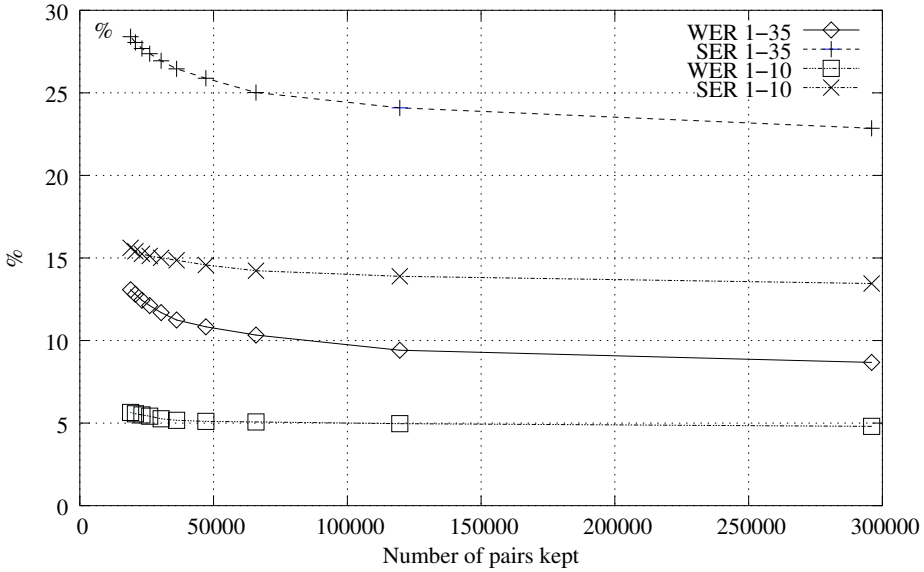

Fig. 2. Performance of the engine as a function of the number of pair of sentences kept in the memory. Each point corresponds to a frequency threshold (from 10 to 1) we considered for filtering the training sentences. These rates are reported for sentences of 10 words or less (1-10) and for 35 words of less (1-35).

Clearly the larger the memory is, the better the performance. The sentence error rate flattens at 23% (13% if measured on short sentences only), while the word error rate approaches 9% (around 5% for short sentences).

In Table 2, we report the best results (obtained for the full memory) that we observed in terms of WER, SER, NIST and BLEU scores. The last two scores were computed on a single reference by the `mteval` program (version v11a), available at `http://www.nist.gov/speech/tests/mt/resources/scoring.htm`. Performances were evaluated on all the sentences of the TEST corpus (FULL column), as well as on the subset consisting of only the sentences that were not found verbatim in the memory (SUBSET column).

We provide the performance as a function of the number of translations returned by the system. When more than one translation is considered, we assume an oracle selecting the best one; in our case, the oracle was choosing the translation with the lowest WER in relation to the reference translation.

	FULL				SUBSET			
$\|n$-best$\|$	WER%	SER%	NIST	BLEU	WER%	SER%	NIST	BLEU
1	9.18	23.56	10.8983	0.8695	22.78	86.80	9.3587	0.6811
2	5.93	16.56	11.1463	0.9071	18.58	84.51	9.7498	0.7211
3	5.02	12.57	11.2740	0.9168	17.13	83.17	9.8823	0.7350
4	4.69	12.04	11.3055	0.9202	16.26	82.54	9.9664	0.7437
5	4.54	11.94	11.3213	0.9219	15.76	81.87	10.0225	0.7493

Table 2. Performance of the engine as a function of the number of translations returned by the system for the full test corpus (FULL) as well as on the subset of the 4641 sentences not seen verbatim in the memory (SUBSET).

Not surprisingly, the performance measured on the full test corpus (FULL) is much better than that measured on previously unseen source sentences (SUBSET). The difference is especially noticable on the SER metric, where 75% of the translations produced in the former case were identical with the reference, while only 15% were in the latter case.

More surprisingly, these figures tell us that our simple approach is a fairly accurate way to translate MÉTÉO sentences, a WER of 9 being several times lower than what we usually observe in "classical" translation tasks. However, we are still below the performance that has been reported in [13]. The author manually inspected a sample of 1257 translations produced by the MÉTÉO2 system and determined that around 11% of them required a correction (minor or not). In our case, and although the SER we measure does not compare directly, we observed on a sample of 36228 translations, that around 24% of them are not verbatim the reference ones.

We analyse in the following section the major errors produced by our approach.

6 Error analysis

We analysed semi-automatically the most frequent errors produced by our process for the 25% of the translations that differed from the reference. We (arbitrarily) selected one of the alignments with the minimum edit distance between the reference translation and the erroneous candidate translation. From this alignment, we identified locus of errors. This process is illustrated in Figure 3 in which an error is indicated by the following notation $\texttt{\textit{SOURCE}} \rightsquigarrow \texttt{\textit{TARGET}}$.

As the classical edit distance between two sequences behaves poorly in the case of word reordering, some errors might not have been analyzed correctly by our process (see the alignment in the last line of Figure 3). However, casual inspection of the errors did not reveal severe problems.

Out of the 8900 errors found at the word level, we manually inspected the 100 most frequent ones (the first ten are reported in Table 3), covering 42.6% of the errors. We found that more than 53% of the errors at the word sequence level

SRC ... *FLURRIES THIS AFTERNOON*
REF ... AVERSES DE NEIGE CET APRES-MIDI
CAN ... AVERSES DE NEIGE CETTE NUIT
ALI = = = **S** **S**

CET APRES-MIDI⇝CETTE NUIT

SRC *WINDS BECOMING LIGHT __DAY1__ MORNING AND THEN ...*
REF VENTS DEVENANT FAIBLES EN MATINEE __DAY1__ PUIS ...
CAN VENTS DEVENANT LEGERS __DAY1__ MATIN PUIS ...
ALI = = **S** **S** **S** **D** =

FAIBLES EN MATINÉE -DAY-⇝LEGERS -DAY- MATIN

SRC ... *AND THEN TO SOUTHWEST __INT2__ THIS EVENING*
REF ... PUIS DU SUD-OUEST A __INT2__ CE SOIR
CAN ... PUIS DIMINUANT DU SUD-OUEST A __INT2__ EN SOIREE
ALI = **I** = = = = **S** **S**

⇝DIMINUANT, CE SOIR⇝EN SOIREE

SRC ... *APPROACHES FROM THE WEST ...*
REF ... A L APPROCHE D UN CREUX VENANT DE ...
CAN ... UN CREUX S APPROCHANT PAR ...
ALI ... **D** **D** **D** **D** = = **S** **S** **I**

A L APPROCHE D⇝, VENANT DE⇝S APPROCHANT PAR

Fig. 3. Illustration of the error detection process on four sentences making use of the edit distance alignment between the reference and the candidate translations. **I** indicates an insertion, **S** a substitution, **D** a deletion, and = the identity of two words. Errors detected are noted as `reference sequence⇝candidate sequence`.

were replacement errors, such as the production of the word LEGER (*light*) instead of the expected word FAIBLE (*light*). Around 30% were suppression errors, i.e. portions of text that have not been produced, but that should have been according to the reference (see A L APPROCHE D⇝ in the fourth alignment of figure 3); the rest (17%) were insertion errors, i.e. a portion of text not found in the reference translation (see ⇝DIMINUANT in the third alignment of figure 3)).

Among the replacement errors, more than 40% are cardinal point substitutions such as *SUD-EST* (*southeast*) instead of *SUD-OUEST* (*southwest*), and more than 7% involve time code substitutions such as HNP/HAP, HNR/HAR, HNE/HAE and HNT/HAT. This means that roughly half of replacement errors could be handled simply by maintaining a small special-purpose lexicon. This also goes for place names that were not dealt with specially in this experiment.

Note that replacement errors are not always synonyms: some are not unrelated CET APRÈS-MIDI⇝CE SOIR (*this afternoon*⇝*tonight*), but others are antonymic as in DIMINUANT⇝AUGMENTANT (*diminishing*⇝*increasing*). This clearly shows the need for post-processing the translation retrieved from the

150	CM.⇝MM.
138	⇝POSSIBILITÉ DE (⇝*chance of*)
134	DE PROBABILITÉ⇝ (*chance of*⇝)
131	TÔT⇝ (*early*⇝)
117	TARD⇝ (*late*⇝)
92	NEIGE⇝PLUIE (*snow*⇝*rain*)
90	DE⇝A
87	SUD-OUEST⇝NORD-OUEST(*southwest*⇝*northwest*)
80	D'OUEST⇝DU NORD-OUEST (*west*⇝*northwest*)
76	⇝TOTALE

Table 3. The 10 most frequent word-sequence errors found with their translation with their number of occurrences in the corpus. The insertion ⇝TOTALE is likely to be an artefact of our TEST corpus, since it often appears in the form ACCUMULATION TOTALE DE __INT___ CM., translated as ACCUMULATION __INT__ CM.

memory. Similarly some insertions and suppressions often modify the semantics of a sentence, as is the case in the fourth error of Figure 3.

7 Discussion

In this paper, we have described a translation memory based approach for the *re*creation of the MÉTÉO system nearly 30 years after the birth of the first prototype at the same university. The main difference between these two systems is the way they were developed. The original system is a carefully handcrafted one based on a detailed linguistic analysis, whilst ours simply exploits a memory of previous translations of weather forecasts that were, of course, not available at the time the original system was designed. Computational resources needed for implementing this corpus-based approach are also much bigger than what was even imaginable when the first MÉTÉO system was developed.

This paper shows that a simple-minded translation memory system can produce translations that are comparable (although not as good) in quality with the ones produced by the current system. Clearly, this prototype can be improved in many ways. We have already shown that many errors could be handled by small specific bilingual lexicons (place names, cardinals, etc.). Our translation memory implementation is fairly crude compared to the current practice in example-based machine translation [2], leaving a lot of room for improvements. We have also started to investigate how this memory-based approach can be coupled with a statistical machine translation system in order to further improve the quality of the translations [12].

8 Acknowledgements

We thank Rick Jones and Marc Besner from the Dorval office of Environment Canada who provided us with the corpus of bilingual weather reports. We are

also indebted to Elliot Macklovitch who provided us with articles describing the MÉTÉO system that we could not have found otherwise. This work has been funded by NSERC and FQRNT.

References

1. Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.
2. Michael Carl and Andy Way, editors. 2003. *Recent Advances in Example-Based Machine Translation*, volume 21 of *Text, Speech and Language Technology*. Kluwer Academic.
3. John Chandioux. 1988. Meteo (tm), an operational translation system. In *RIAO*.
4. J. Coch. 1998. Interactive generation and knowledge administration in multimeteo. In *Ninth International Workshop on Natural Language Generation*, pages 300–303, Niagara-on-the-lake, Ontario, Canada.
5. G. Foster, S. Gandrabur, P. Langlais, E. Macklovitch, P. Plamondon, G. Russell, and M. Simard. 2003. Statistical machine translation: Rapid development with limited resources. In *Machine Translation Summit IX*, New Orleans, USA, sep.
6. E. Goldberg, N. Driedger, and R. Kittredge. 1994. Using natural language processing to produce weather forecasts. *IEEE Expert 9*, 2:45–53, apr.
7. Annette Grimaila and John Chandioux, 1992. *Made to measure solutions*, chapter 3, pages 33–45. J. Newton, ed., Computers in Translation: A Practical Appraisal, Routledge, London.
8. W. John Hutchins and Harold L. Somers, 1992. *An introduction to Machine Translation*, chapter 12, pages 207–220. Academic Press.
9. R. Kittredge, A. Polguère, and E. Goldberg. 1986. Synthesizing weather reports from formatted data. In *11th. International Conference on Computational Linguistics*, pages 563–565, Bonn, Germany.
10. Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Second Conference on Human Language Technology Reasearch (HLT)*, pages 127–133, Edmonton, Alberta, Canada, May.
11. Philippe Langlais, Michel Simard, and Jean Véronis. 1998. Methods and practical issues in evaluating alignment techniques. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL)*, Montréal, Canada, August.
12. Thomas Leplus, Philippe Langlais, and Guy Lapalme. 2004. A corpus-based Approach to Weather Report Translation. *Technical Report*, University of Montréal, Canada, May.
13. Elliott Macklovitch. Personnal communication of results of a linguistic performance evaluation of METEO 2 conducted in 1985.
14. S. Nießen, S. Vogel, H. Ney, and C. Tillmann. 1998. A dp based search algorithm for statistical machine translation. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL) and 17th International Conference on Computational Linguistics (COLING) 1998*, pages 960–966, Montréal, Canada, August.
15. Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press. 270 p.