Translation of sublanguages by subgrammars

Julien Gosme, Yves Lepage and Adrien Lardilleux
GREYC
University of Caen Basse-Normandie
Firstname.Lastname@info.unicaen.fr

Abstract

This paper discusses the performance of two data-driven translation methods for the translation of a very constrained sublanguage: dates. As a first result, we show that an example-based method is outperformed by a statistical method for the translation of dates from Chinese into English when small random training corpora are used: 750 random examples suffice to translate almost perfectly a corpus of 4,018 dates for both methods. As a second result, we prove that 58 dates theoretically suffice to translate the same corpus of 4,018 dates perfectly and we verify this fact experimentally with an example based method, while a statistical method fails at translating 345 dates in the 4,018 dates to translate.

1 Introduction

It is well known that machine translation of sublanguages is easier than machine translation of general texts. For example, TAUM-METEO [3] was designed to translate weather forecast bulletins and alerts from English into French and vice-versa. Translations produced by this system are assessed as almost perfect. By design, this system cannot translate out-of-domain texts.

According to Kittredge [4], a sublanguage is a restriction in lexicon and syntactic constructions. Harris [2] goes one step further when he states:

> ”[...] The initial phase of sublanguage analysis establishes a direct relationship between surface sentence forms and [their] semantic representation [...].”

One can expect less out-of-vocabulary words from machine translation systems designed for a specific sublanguage. In addition, sublanguages typically present less polysemy.

In an example-base setting, we shall call a corpus of examples specially designed and dedicated for the translation of a sublanguage an *example subgrammar*. In this paper, we shall investigate the properties that such a subgrammar should possess in order to help in the translation of an extreme case of constrained sublanguages: that of dates. The sublanguage of dates is indeed enumerable for a given range of dates in any human language and its semantics is direct. Two data-driven machine translation systems will be compared: a statistical one and an example-based one.

The paper is organized as follows: Section 2 deals with the determination of the optimal size of subgrammars experimentally and analytically. Section 3 describes a specialization of an example-based translation system for dates. Section 4 assesses this example-based translation system. Section 5 suggests ways to automatically construct example subgrammars based on a corpus of translation examples.

2 Optimal size of subgrammars

2.1 An experimental approach

In this section, we are concerned with the problem of finding the optimal size of a corpus of examples designed as a training corpus for translation. For this purpose, we enumerate all dates from Saturday 1st January 2000 to Friday 31st December 2010 in English and in Chinese. There are 4,018 such dates. We

Mikel L. Forcada, Andy Way (eds.)
Proceedings of the 3rd Workshop on Example-Based Machine Translation, p. 77–84
Dublin, Ireland, November 2009

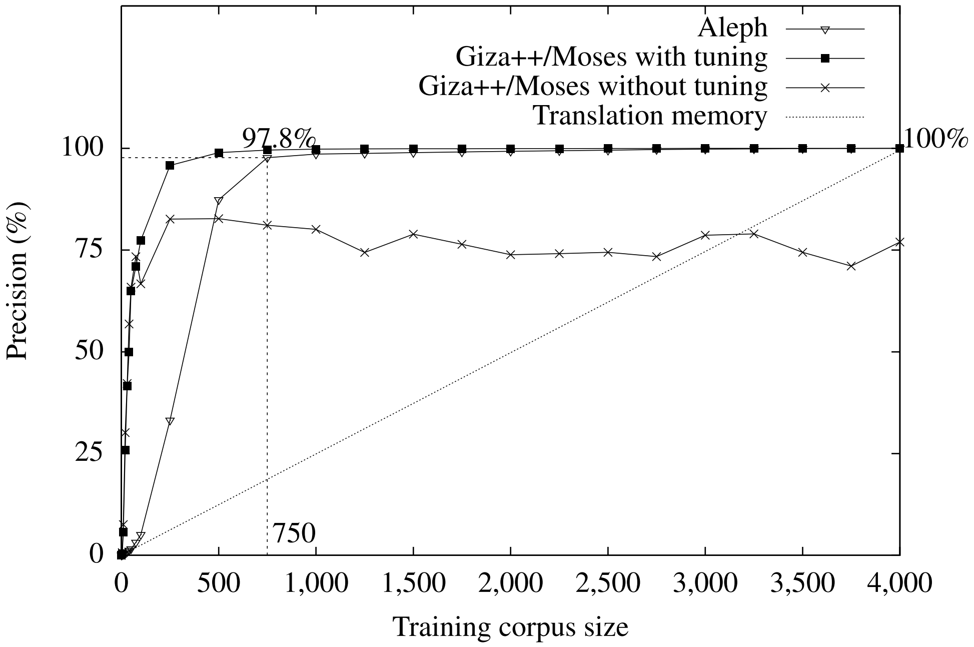


Figure 1: Comparison between three machine translation methods and a baseline for the translation of dates from Chinese into English. Moses with a tuning phase and Aleph translate more than 97% dates with example corpora larger than 750 pairs of sentences.

prepare training corpora of different sizes by sampling the corpus. Starting from a corpus of 4,000 samples extracted from the 4,018-dates corpus, we extract in a recursive way smaller sub-corpora randomly by subtracting 250 sentences at each step. For smaller sizes, we subtract 10 sentences until exhaustion.

We compare three different machine translation methods:

1. an example based method with the machine translation system Aleph as described in [6];
2. a statistical method implemented by combining off-the-shelf tools: Moses [5] using Giza++ [7] for alignment. Distortion limit is set to 4 (maximum value) to allow reordering;
3. a statistical method similar to the one described above with a phase of tuning using MERT [8] prior decoding.

In addition, the three previous machine translation methods will be compared to a baseline system: a translation memory which contains exactly the training corpus.

All the 4,018 Chinese dates from the original corpus compose the test corpus and the 4,018 corresponding English dates compose the reference corpus. Thus, there is one reference date per test date. Because dates are way too short, we choose not to use the BLEU metric [9] to compare systems results. Rather, the criterion we use for comparison between all translation methods is precision, defined as the percentage of translation hypotheses, exactly matching the English reference. The configuration of Moses with tuning resorts to 10-fold cross-validation. The development set is taken from the training corpus, hence shortly reducing the number of training sentence pairs. For each sub-corpus size, this experiment is repeated 10 times and an average is taken. The results are shown in Figure 1.

An amount of 750 examples of dates suffices to translate more than 97% of a corpus of 4,018 dates with a tuned statistical approach or an example-based system. Larger corpora lead to insignificant improvements.

Moses with tuning needs only training corpora of sizes greater than 250 to translate more than 96% of the whole corpus. The Aleph system needs 750 examples to reach 97.8% precision. For both systems, their precisions increase regularly and slowly up ti 100% with 4,000 examples and are not distinguishable after 1,000 examples.

Moses without tuning exhibits a chaotic behavior and even performs worse than a translation memory with more than 3,250 examples.

2.2 An analytical approach

In this section, we prove that even 750 examples are not necessary: indeed, 58 dates theoretically suffice to translate the corpus of 4,018 dates described in the previous section.

In Chinese, dates are composed of the following four parts in this order:

- the year (e.g., 二〇〇一年 means 2001);
- the month (e.g., 四月 means April);
- the day of the month (e.g., 一日 means 1st of the month);
- the week-day (e.g., 星期一 means Monday).

In British English, dates are written in the reverse order, e.g. “Thursday 1st January 2009”.

As four parts can clearly be identified, we propose to write an example subgrammar for each part of a date.

An example subgrammar is defined as a set of translations given a left context and a right context. For instance, we may choose the subgrammar for the description of month translations to get “Monday 1st” as a left context and “2001” as a right context. In a subgrammar, only the considered part vary, the contexts being left unaltered.

Example subgrammars for the translation of dates are given in Table 1(a) for years, in Table 1(b) for months, in Table 1(c) for days of months, and in Table 1(d) for week-days. Altogether, these 4 example subgrammars are composed of 58 unique examples: 11 for years, 12 for months, 31 for days, and 7 for week-days, minus three common examples. We call *prototype* the example shared by the 4 subgrammars. As these 4 subgrammars exhaustively describe all phenomena, 58 dates theoretically suffice to translate the entire corpus of 4,018 dates. The proof is as follows:

Proof. The proof is in 4 steps and uses the translation by analogy principle. In each step we prove that we can translate the relevant part of any Chinese date of the form YMDW using the relevant example subgrammar. Let YMDW the date to be translated. With an example from the subgrammars and the prototype $Y_0M_0D_0W_0$, we can always form the following analogical equation: $< example > \; : \; < prototype > \; :: \; YMDV \; : YM_0DW$. Similar applications in any order in sequence on the input necessarily lead to $Y_0M_0D_0W_0$ the translation of which is known: $\widehat{W_0}\widehat{D_0}\widehat{M_0}\widehat{Y_0}$. In the target language, analogical equations are solved in the reverse order; this process replaces step by step each of the 4 parts to finally produce $\widehat{W}\widehat{D}\widehat{M}\widehat{Y}$, each $\widehat{W}$, $\widehat{D}$, $\widehat{M}$, and $\widehat{Y}$ coming from each relevant example subgrammar. □

3 An example-based translation system using subgrammars

In this section, we describe a specific subgrammar translation system derived from the Aleph system. This system focuses on recursions which take place in the system. The main difference is that our derived system considers one subgrammar per recursion depth. Since subgrammars describe independent

Table 1: Example subgrammars for the translation of years in Table (a), months in Table (b), days of the month in Table (c) and the week-day in Table (d). The example in common, called *prototype*, is "二〇〇一年 四月 一日 星期一 ↔ Monday 1st April 2001", this example is underlined in all subgrammars.

$Y \leftrightarrow \widehat{Y}$
二〇〇〇年 ↔ 2000
二〇〇一年 ↔ 2001
二〇〇二年 ↔ 2002
二〇〇三年 ↔ 2003
⋮

(a) Contexts: Y 四月 一日 星期一 ↔ Monday 1st April $\widehat{Y}$

$M \leftrightarrow \widehat{M}$
一月 ↔ January
二月 ↔ February
三月 ↔ March
四月 ↔ April
⋮

(b) Contexts: 二〇〇一年 M 一日 星期一 ↔ Monday 1st $\widehat{M}$ 2001

$D \leftrightarrow \widehat{D}$
一日 ↔ 1st
二日 ↔ 2nd
三日 ↔ 3rd
四日 ↔ 4th
⋮

(c) Contexts: 二〇〇一年 四月 D 星期一 ↔ Monday $\widehat{D}$ April 2001

$W \leftrightarrow \widehat{W}$
星期一 ↔ Monday
星期二 ↔ Tuesday
星期三 ↔ Wednesday
星期四 ↔ Thursday
⋮

(d) Contexts : 二〇〇一年 四月 一日 E ↔ $\widehat{E}$ 1st April 2001

phenomena, it is possible to translate each phenomenon independently (one per recursion step). A consequence and an advantage of this view is that translation time is significantly reduced.

3.1 System description

Translation is performed in two steps:

1. collecting the examples needed for translation from the subgrammars;
2. translating the input date by combining the collected examples.

The decomposition into 4 subgrammars reduces the number of examples considered simultaneously. For each example in a subgrammar, only the considered phenomenon varies, the contexts being constant. During the step of collecting examples from subgrammars, the nearest example is selected among all examples. Equal contexts in a given subgrammar, ensure the selection of the example which leads to the best translation.

3.2 Vauquois' triangle

The subgrammar translation system has some similarities with Vauquois' triangle [12] (see [1] for a detailed representation).

Each translation can be represented as a trapezoid which is a part of a triangle (see Figure 2). The first step in our system is similar to the analysis in Vauquois' triangle. The left side is ascending and starts from the input sentence. As we go along the edge, we get closer to the prototype sentence. Vauquois' transfer is equivalent to the step of recursion (the upper arrow of the trapezoid). The same method applies on the resulting sentence, hence the name step of recursion. The third step is similar to generation in

Vauquois' triangle. This step consists in another analogical equation to solve in the target language. Finally the constructed translation closes the trapezoid (bottom edge) and allows recursion steps to take place.

$$\begin{tikzcd}
 & & P \to \widehat{P} \arrow[dr, dashed] & & \\
 & S_{n+1} \arrow[ur, dashed] \arrow[rr] & & \widehat{S_{n+1}} \arrow[dr, "\widehat{E_n} : \widehat{P} :: \widehat{S_n} : \widehat{S_{n+1}}"] & \\
S_n \arrow[ur, "E_n : P :: S_n : S_{n+1}"] \arrow[rrrr] & & & & \widehat{S_n}
\end{tikzcd}$$

Figure 2: Similarities between Vauquois' triangle and our translation system.

3.3 Step by step translation in our system

Figure 4 (see last page) details the translation of the Chinese input S_0="二〇〇七年 十一月 十六日 星期五" with our system.

The computation starts in Chinese and transforms the input date to the Chinese prototype P step by step. A different subgrammar is considered at each recursion depth n. First, the closest example E_0 is taken from the subgrammar 1(d). The analogical equation $E_0 : P :: S_0 : x$ is solved and yields the solutions S_1. If the equation has no solution then $S_1 = S_0$.

The recursion stops when $S_n = P$. The Chinese date S_4 ="六七八九年 四月 一日 星期一" equals the Chinese prototype P. Therefore, translations are known for P, E_3, and S_4. They are noted $\widehat{P}$, $\widehat{E_3}$, and $\widehat{S_4}$. The computation switches to English starting with the analogical equation $\widehat{E_3} : \widehat{P} :: x : \widehat{S_4}$. Solutions of this equation are translations of S_3: $x = \widehat{S_3}$. Computation continues until $\widehat{S_0}$ has been computed.

Finally, $\widehat{S_0}$ is a translation of the input date S_0. In this example $\widehat{S_0}$ ="Friday 16th November 2007".

4 Translation experiments

In this section, we experimentally evaluate our subgrammar system to check whether it can really translate the set of 4,018 dates. At the same time we shall compare this system with a statistical machine translation system.

Following the experiment in Section 2.1, we use the set of 4,018 dates from Saturday 1st January 2000 to Friday 31st December 2010 (Chinese for test and English for reference).

We compare our system with the statistical machine translation system Moses. With default settings, Moses disallows word reordering (distortion). Thus we compare 5 configurations of Moses by varying the distortion limit from 0 to 4 words. This experiment is performed twice: with and without tuning.

All systems are compared with TER [10] and precision (as in Section 2.1). All systems use the subgrammars listed in Tables 1(a), 1(b), 1(c), and 1(d).

Results are shown in Table 2. As was expected, and by design, our system is able to translate all dates with 100% precision. As for Moses, the best results are obtained when the distortion limit is set to 4 or higher. Because English dates are written in the opposite direction of Chinese dates, and because dates are composed of 4 terms, it is natural for the distortion limit to be at least 4. Moses(2-3) get a TER score of 0.25 because a distortion limit of 2 or 3 allows reordering of some words. On the other hand, a distortion limit of 0 or 1 disallows reordering, resulting in a TER value of 0.75 (worst score for 4 words).

Table 2: Comparison of our system with Moses.

	Without tuning		With tuning	
System (Distortion limit)	TER	Precision (%)	TER	Precision (%)
Moses(0)	0.75	0	0.68	0.20
Moses(1)	0.75	0	0.68	0.20
Moses(2)	0.25	0	0.34	0.05
Moses(3)	0.25	0	0.33	0.02
Moses(4-. . .)	0.02	92	0.09	69.00
Subgrammar system	**0.00**	**100**	—	—
Translation memory	0.67	1	—	—

Moses with the distortion limit set to 4 translates all 4,018 dates except 345. Those 345 dates appear to be in April. Since the dates of our subgrammars derive from Monday 1st April 2001, April appears in most of our examples and thus introduces a bias compared to a corpus of random dates as used in Section 2.1. Statistical machine translation systems appear to be incompatible with the subgrammars we introduced, because the goal of subgrammars is to eliminate redundancy in the training data, which as a consequence, also eliminates balance in the data.

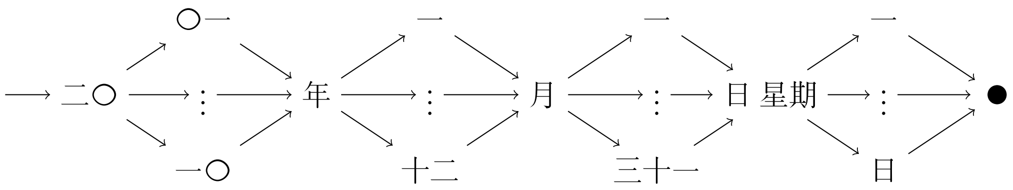
Note that Moses(4-. . .) obtained better scores without tuning than with tuning, the reason for this being that training corpora are shorten by 10% (for development sets).

5 Future research: automatic acquisition of example subgrammars

The experiments described in Sections 2.1 and 2.2 show that the automatic constitution of subgrammars based on a corpus of translation examples is a problem of generalization. The tuning step in a statistical approach actually performs a generalization. In the case of an example-based method, better results were obtained with generalized examples such as the example subgrammars introduced in Section 2.2.

Dates are an example of a regular language. From our 4,018 Chinese examples, and using techniques of automaton minimization we obtain the automaton drawn on Figure 3 which distinguishes 4 variable parts. In actual corpora, only a certain number of dates appear, from which the problem is to obtain the kind of automaton representation as in Figure 3 or subgrammars as in Section 2.2.

Figure 3: Minimized automaton obtained from our corpus of 4,018 dates from which 4 variable parts are clearly distinguishable.



The unsupervised learning algorithm ADIOS (Automatic DIstillation Of Structure) [11] may be a way to acquire such subgrammars from a corpus of examples thanks to its operation of generalization.

6 Conclusion

In this paper, we assessed an example-based machine translation system on tiny constrained sublanguages. The sublanguage chosen as an illustration consists in the dates in Chinese and English, ranging from Saturday 1st January 2000 to Friday 31st December 2010. In our settings, Chinese dates have exactly one translation in English and reciprocally. We found experimentally that the statistical machine translation system Moses with a tuning step and the example-based machine translation system Aleph need only 750 examples of translation randomly chosen among the 4,018 to translate almost all dates in this range. The same experiment shows that Moses outperforms Aleph for training corpora smaller than 750 pairs of sentences.

In a second step, we proved analytically that 58 translation examples carefully chosen suffice to translate all 4,018 dates. These 58 dates all derive from a single example called prototype of which one part vary at a time. Example subgrammars are constituted by varying one part of the prototype at a time.

In addition, we described a modified version of the Aleph system which takes advantage of these subgrammars to perfectly translate the entirety of sublanguage. In contrast, 8% of the corpus is not properly translated by Moses without tuning. The performance of Moses is even worse when tuned. By construction, our subgrammars contain all examples required to translate the 4,018 dates, which makes the corpus unbalanced, a drawback for statistical approaches.

In the future we will focus on the problem of automatic construction of subgrammars from actual examples. This problem can be seen as a problem of generalization in the terms of ADIOS.

References

[1] C. Boitet. Corpus pour la TA : types, tailles et problèmes associés, selon leur usage et le type de système. *Revue Française de Linguistique Appliquée*, (121):25–38, 2007.

[2] Z. Harris, M. Gottfried, T. Ryckman, P. Mattick Jr, A. Daladier, T.N. Harris, and S. Harris. *The form of information in science: analysis of an immunology sublanguage*. Kluwer Academic Publishers, 1989.

[3] P. Isabelle. Machine translation at the TAUM group. *Machine Translation: The State of the Art*, pages 247–277, 1987.

[4] R. Kittredge. The significance of sublanguage for automatic translation. *Machine Translation: Theoretical and methodological issues*, pages 59–67, 1987.

[5] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Ze ns, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In *ACL 2007*, Prague, Czech Republic, June 2007.

[6] Yves Lepage and Etienne Denoual. Purest ever example-based machine translation: detailed presentation and assessment. *Machine Translation Journal*, 19:251–282, 2005.

[7] F.J. Och and H. Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, 2003.

[8] Franz Josef Och. Minimum error rate training in statistical machine translation. In *ACL 2003*, pages 160–167, Sapporo, Japan, July 2003. Association for Computational Linguistics.

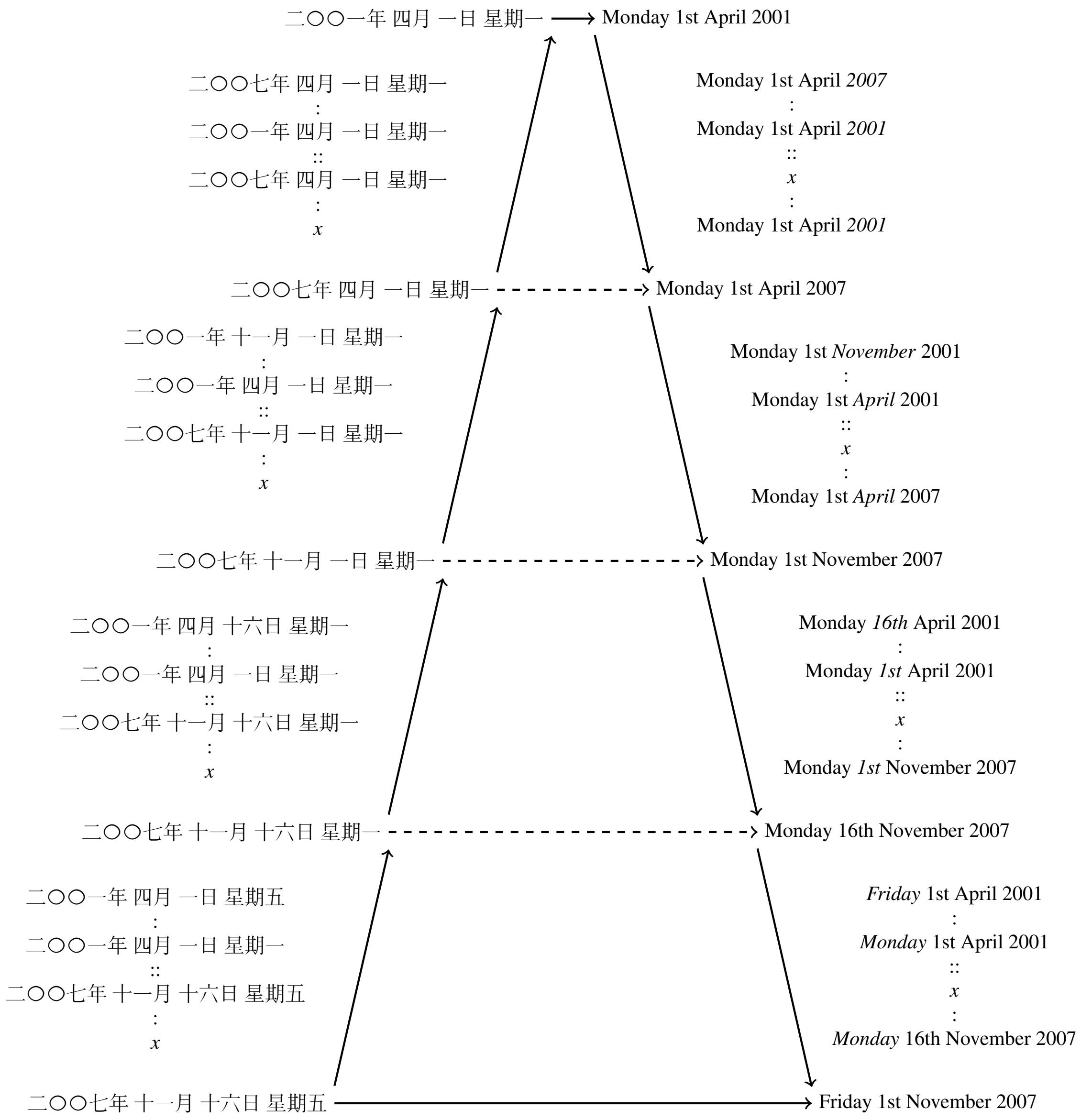
[9] K. Papineni, S. Roukos, T. Ward, and W.J. Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *ACL 2002*, pages 311–318, 2002.

[10] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. A study of translation edit rate with targeted human annotation. In *AMTA 2006*, pages 223–231, 2006.

[11] Z. Solan, E. Ruppin, D. Horn, and S. Edelman. Automatic acquisition and efficient representation of syntactic structures. *Advances in Neural Information Processing Systems*, pages 107–116, 2002.

[12] B. Vauquois and C. Boitet. Automated Translation at Grenoble University. *Computational Linguistics*, 11(1):28–36, 1985.

Figure 4: Detailed execution of the subgrammar translation system deriving from Aleph. The input date is ”二〇〇七年 十一月 十六日 星期五”. The path composed of dashed arrows shows the steps of translation of the system. Bidirectional arrows correspond to translation equivalences. The prototype P is constant, S_n are input dates. E_n is the closest date to the input date S_n and S_{n+1} is the solution of the analogical equation $E_n : P :: S_n : x$. In the target language (English in this case), $\widehat{E_n}$ is the translation of E_n, $\widehat{P}$ is the translation of P, $\widehat{S_{n+1}}$ is the input date and $\widehat{S_n}$ is the solution of the analogical equation $\widehat{E_n} : \widehat{P} :: x : \widehat{S_{n+1}}$.



二〇〇一年 四月 一日 星期一
Monday 1st April 2001
二〇〇七年 四月 一日 星期一 : 二〇〇一年 四月 一日 星期一 :: 二〇〇七年 四月 一日 星期一 : x
Monday 1st April *2007* : Monday 1st April *2001* :: x : Monday 1st April *2001*
二〇〇七年 四月 一日 星期一
Monday 1st April 2007
二〇〇一年 十一月 一日 星期一 : 二〇〇一年 四月 一日 星期一 :: 二〇〇七年 十一月 一日 星期一 : x
Monday 1st *November* 2001 : Monday 1st *April* 2001 :: x : Monday 1st *April* 2007
二〇〇七年 十一月 一日 星期一
Monday 1st November 2007
二〇〇一年 四月 十六日 星期一 : 二〇〇一年 四月 一日 星期一 :: 二〇〇七年 十一月 十六日 星期一 : x
Monday *16th* April 2001 : Monday *1st* April 2001 :: x : Monday *1st* November 2007
二〇〇七年 十一月 十六日 星期一
Monday 16th November 2007
二〇〇一年 四月 一日 星期五 : 二〇〇一年 四月 一日 星期一 :: 二〇〇七年 十一月 十六日 星期五 : x
Friday 1st April 2001 : *Monday* 1st April 2001 :: x : *Monday* 16th November 2007
二〇〇七年 十一月 十六日 星期五
Friday 1st November 2007