Using patterns for Machine Translation (MT)

Stella Markantonatou, Sokratis Sofianopoulos, Vassiliki Spilioti, George Tambouratzis, Marina Vassiliou, Olga Yannoutsou*

Institute for Language and Speech Processing, Artemidos 6; 151 25 Athens (Greece)
{marks | s_sofian | vspiliot | giorg_t | mvas | olga}@ilsp.gr

Abstract. In this paper an innovative approach is presented for MT, which is based on pattern matching techniques, relies on extensive target language monolingual corpora and employs a series of similarity weights between the source and the target language. Our system is based on the notion of 'patterns', which are viewed as 'models' of target language strings, whose final form is defined by the corpus.

1. Introduction

With this work, we further explore the ideas tested within the METIS-I[1] system (Dologlou et al. 2003) which proved the feasibility of the innovative idea that sound translations could be received with hybrid MT that relied on monolingual corpora – rather than parallel ones – and flat bilingual lexica. This is the main difference between METIS systems and corpus-based approaches (EBMT, SMT) which rely on bilingual corpora. For corpus-based MT approaches, which have taken the lead from rule-based ones (Hutchins 1995), the basic resources, i.e. parallel corpora, are scarce. Such corpora are rare and available for the very widely spoken languages only. In addition, they quite often represent a certain register or sublanguage. Efforts to face the problem have focused on reducing the size of the required parallel corpus (Al Onaizan (2000), Brown (2003)). By resorting to monolingual corpora only, the METIS projects pursue a radically different solution to the problem of scarcity of resources. However, METIS-I too faced a serious problem of sparseness of data as it could manipulate only sentences as units. In METIS-II, the framework of the present work (Markantonatou et al. 2005), material at sub-sentential level, namely chunks, is exploited to generate translations.

The great promise with corpus-based approaches lies in that 'hard-to-manipulate' linguistic information can be induced from the corpus rather than being explicitly represented with a constantly growing collection of rules. The syntactic and semantic preferences of words (one of the reasons why the number of rules tends to explode in both hand-crafted and tree-bank induced grammars (Gaizauskas 1995)) constitute a large part of the implicit information provided by the corpus. A similar argument can be made about word order. Thus, work on (various approaches to) corpus-based MT aimed at making do without resorting to any expensive linguistic resources such as (rich) computational lexica and grammars (Nagao 1984, Brown 1990). However, it has become evident that some amount of linguistic knowledge is necessary (see, for instance Popowich (2005) for the case of SMT and Carl & Way (2003) for various uses of linguistic resources in Example-Based MT). Actually, nowadays, investigation of hybrid systems combining easy-to-obtain resources from all MT paradigms, rule-based included, is considered a very promising path of research in the field ((Nirenburg & Raskin (2004), Thurmair (2005)).

[1] METIS was funded by EU under the FET Open Scheme (METIS-I, IST-2001-32775), while METIS-II, the continuation of METIS, is being funded under the FET-STREP scheme of FP6 (METIS-II, IST-FP6-003768). The assessment project METIS ended in February 2003, while the second phase started in October 2004 and has a 36 month duration.

In the work presented here, an innovative hybrid approach is adopted, which relies on target language (TL) corpus information at sub-sentential level and employs pattern matching techniques. Many efforts to exploit sub-sentential evidence are reported in state-of-the-art MT and range from n-gram approaches in SMT (Ney 2005) to sophisticated parsers' output (Way 2003) and template alignment (McTait 2003) in EBMT. The pattern matching technique we present here uses the monolingual corpus as a source of TL patterns and as a repository of implicit information, which is exploited to resolve issues related with lexical affiliations in the TL (co-occurrence tendencies, argument selection) and to capture language-dependent properties such as word order.

2. Patterns

Several researchers in the corpus-based MT paradigm have reported on the use of patterns. However, these patterns differ from the patterns employed in the work presented here. Lepage (1997) employs sequences of words to improve matching with the source language (SL) side of the parallel corpus. Best matching scores are achieved when long SL strings of the parallel corpus are identical with strings in the input sentence. No operations on strings are foreseen. McTait (2003), Brown (2003) and Kitamura (2004) (among others) create patterns, namely sequences of words and variables standing for sequences of words, both for matching on the SL side and for generating translations. In the work presented here the term 'pattern' is not used in any of the ways presented above for two reasons: (a) there are no parallel corpora and there is no direct matching of the SL string with strings in the same language and (b) more important, patterns are not viewed as fixed strings with or without slots for variables but as 'models' of TL strings, which receive their final form only after the corpus has been consulted. Consultation of the corpus is performed with pattern matching techniques.

The intuition behind patterns as used in the work presented here is simple. The SL structure consists of a verb and satellite chunks which are either arguments of the verb or modifiers denoting time, place or manner. In the general case, we would like to recover in the TL the verbal meaning and the meaning conveyed with the satellite chunks. For instance, if an event is described in the SL involving two participants and information about time and place, we would like the translation to report about the same event with the same number of participants and the same information about time and place. Crucially, however, we do not require that all these meaning components are of the same syntactic status across the language pair. This is achieved with the mechanism of the pattern matching algorithm, which employs a set of similarity weights (see Section 2.3) and allows for similar grammatical and syntactic categories in addition to identical ones. In this sense an AdjP may match with an AdjP, an NP or a PP in reduced similarity order.

2.1. Patterns in SL and TL

Patterns are generated by the output of the chunkers used for both languages and are formed by chunks and their respective constituents. Depending on the phase of the matching algorithm different types of pattern are used, as the system concentrates on different types of information. It must be noted, however, that only a very small number of pattern types is required. Thus, for both the SL and the TL only three types of pattern are used: the Clause Pattern, the VG Pattern and the PP Pattern.

Clause Pattern

(PP* token*)* VG (PP* token*)* *[where 'token' refers to adverbials and punctuation]*

The **Clause pattern** describes the overall structure of a clause: the verbal group head and the number, labels and heads of the chunks (if any exist).

The **VG pattern** describes the verb group. Other tokens such as adverbs for example, if found within the verb phrase will be part of it, while if found in isolation, are not considered to form a pattern and will be treated in a different way.

The **PP pattern** describes both prepositional and noun chunks in terms of their constituent tokens. The generalisation here is that a noun chunk can be represented as a preposi-

tional one with an empty prepositional head. This representation captures phrase category mismatches between SL and TL of the sort exemplified in (1).

1.[$_{pp}$ ∅ [$_{np\_nom}$ *ο σκύλος*]] [$_{vg}$ *μπήκε*] [$_{pp}$ ***στο*** [$_{np\_acc}$ ***δωμάτιο***]]

[$_{pp}$ ∅ [$_{np1}$ *the dog*]] [$_{vg}$ *entered*] [$_{pp}$ ∅ [$_{np2}$ ***the room***]][2]

2.2. Pattern acquisition

This is a hybrid approach, because pattern acquisition is rule-based: already existing and rather trivial tools are used for both the SL and TL and include taggers, lemmatisers and chunkers. Certainly, adjustments had to be made to both the SL and TL tools to improve compatibility of the resulting patterns.

The TL corpus is processed off-line once and then stored in a relational database of TL patterns containing (a) clause patterns indexed on the basis of their main verb and the number of their chunks and (b) PP patterns classified according to their head.

The pattern derived from the SL input, the "TL-like pattern" from now on, is created in real time. The SL input is tagged, lemmatised, chunked and fed as input to a bilingual flat dictionary. All tokens from the SL string (2) are looked up in the lexicon and multiple translation equivalents are derived (3). No score is related with the multiple translations. It must be stressed that one of the advantages of the pattern matching approach presented here is that it does not rely on frequency information: as opposed to statistical approaches, the pattern matching one does not miss rare occurrences and combinations of words or patterns.

2.[$_{ppgof}$ ∅ [$_{np\_nm}$ *Ο υπουργός*] [$_{np\_ge}$ *Οικονομικών*]] [$_{vg}$*διέλυσε*] [$_{ppgof}$ ∅ [$_{np\_ac}$ *τη συνάντηση*] [$_{np\_ge}$ *της επιτροπής*]] [$_{ppgof}$ ***για*** [$_{np\_ac}$ *την κακοποίηση*] [$_{np\_ge}$ *ανηλίκων*]]

(literal translation: The Finance Minister broke up the committee meeting about child abuse)[3]

3.[$_{ppgof}$ ∅ [$_{np\_nm}$ *The minister / secretary*] [$_{np\_ge}$ *Finance / economics*]] [$_{vg}$ *break up / dissolve*] [$_{ppgof}$ ∅ [$_{np\_ac}$ *meeting / encounter*] [$_{np\_ge}$ *commission / committee*]] [$_{ppgof}$ *for / about* [$_{np\_ac}$ *abuse*] [$_{np\_ge}$ *child / juvenile*]]

The multiple TL-like patterns obtained are fed to the core translation engine to match them against respective patterns in the TL corpus. Thus, in our approach, rather than asking, as Nagao (1984) and the EBMT paradigm did, 'tell me how you have translated it and I will repeat the translation', we require that the algorithm, which we provide with TL-like strings, exploits corpus information and elicits grammatical strings.

2.3. Pattern matching

As mentioned before, METIS-II maps TL-like patterns onto patterns retrieved from the monolingual TL corpora. By addressing this matching problem as a general, weighted assignment problem, METIS-II manages to resolve translation issues without resorting to linguistic rules.

Mapping is carried out by comparing patterns in both languages and assigning scores. The degree of similarity across patterns is revealed on the basis of appropriate information depending on the types of pattern compared. Scores are calculated with the use of a series of weights[4], which provide information regarding the similarity of tags, tokens, lemmata and chunk labels. Chunk labels denote categorical status apart from the label NP1 for the TL which denotes a pre-verbal nominal chunk adjacent to the verb group. For example, a tag similarity weight with a value of 0 indicates that the two tags involved cannot be considered as 'matching' (e.g. a verbal tag will never map onto a prepositional tag), while a value of 1 would mean that 'matching' is ideal. Weights are used by the assignment algorithm in order to achieve the optimum mapping. Thus the system manages to correct the word order and delete/insert tokens. In the following section we use an example to illustrate the overall procedure.

[2] NP1 and NP2 are chunk labels indicating the position of the TL PP patterns in relation to the VG pattern.

[3] Heads of PP patterns are marked with bold.

[4] For a discussion of the formulas used for score calculation, see Markantonatou et al. (2005) and Tambouratzis et al. (2006).

3. Patterns

In four distinct steps, the pattern-matching algorithm proceeds gradually from wider patterns to narrower ones, ensuring that the largest continuous piece of information is retrieved as such, while mismatching areas are identified. We will illustrate the procedure by using the SL sentence in (4), where the clause pattern has a VS order, which is ungrammatical for English declarative, non-emphatic clauses:

4.Συνήθως [$_{vg}$ διαρκούν] για ώρες [$_{pp}$ ∅ [$_{np\_nm}$ οι εβδομαδιαίες συναντήσεις] [$_{np\_ge}$ των πιο βαρετών ανθρώπων]

(literal translation: Usually last for hours the weekly meetings of the most boring people)

We expect the system to produce string (5) which will then be fed to a morphological generator for English (not yet implemented):

5.[$_{pp}$ ∅ [$_{np1}$ the weekly meeting] [$_{pp}$ of [$_{np2}$ the most boring people]] usually [$_{vg}$ last] for hour

SL string (4) is tagged, lemmatised and chunked and the resulting TL-like pattern is fed to the system. At the first step the algorithm delimits the matching process within the clause boundaries. Therefore, the TL clause database is searched for clause patterns similar to the TL-like one in terms of the verbal head and the number of contained chunks, which should equal or exceed by up to 2 the chunk number of the TL-like pattern. The best matching clause retrieved from the TL corpus at this step is given in (6):

6.One charge of the battery lasts for hours, even at top speed,

At the second step, the retrieved TL clause patterns are compared with the TL-like one at a lower level, namely, with respect to the type and head of the chunks contained. The degree of the patterns' functional and lexical similarity is determined and the establishment of the chunk pattern order is achieved. Table (1) illustrates how the VS order of the TL-like pattern is fixed to the right SV order illustrated in (5), by relying on information implicit in the corpus-retrieved sentence (6).

More specifically, the system manages to establish the correct word order, after matching a TL-like PP pattern in nominative (np_nm) with a TL PP pattern (NP1) that precedes the verb (Table 1). This matching is achieved by employing the respective similarity weight (Table 2), whose value is 1, when comparing the chunk labels np_nm and NP1, thus enabling the algorithm to establish the structure (the SV order) in the final translation, before handling the lexical differences between the heads and the tokens at a next step.

Translated Sentence:	usually , the weekly meeting the most boring people last for hour				
Corpus Sentence:	One charge of the battery lasts for hours, even at top speed ,				
Score = 83.739136%	pp([-{-}] **np_nm**(the{AT0} adjp([weekly{AJ}]) [meeting{NN}]))	pp ([of{-PRF}] np_ge(the{AT0} adjp(most{AV0} [boring{AJ}]) [people{NN}]))	vg([last {VV}])	pp ([for{PRP}] np_ac ([hour{NN}]))	PAD
PP([-{-}] **NP_1**(ADJ([one {CRD}]) [charge{NN1}]))	**79%**	61%	0%	61%	20%
PP([of{PRF}] NP_2(the{AT0} [battery{NN1}]))	40%	**79%**	0%	78%	20%
VG([last{VVZ}])	0%	0%	**100%**	0%	20%
PP([for{PRP}]NP_2([hour{NN 2}]))	40%	78%	0%	**100%**	20%
PP([at{PRP}]NP_2(ADJ([top{ AJ0}])[speed {NN1}]))	40%	78%	0%	78%	**20%**

Table 1. Clause comparison based on chunk labels & chunk heads.

NP_NM	NP_1	**1**
NP_NM	NP_2	0.1

Table 2. Chunk label comparison similarity weights.

At the third step, the pattern matching algorithm performs a detailed comparison between the tokens contained in the TL chunk patterns and the respective TL-like ones, in order to establish degrees of lexical similarity and thus decide upon whether the TL chunk patterns will be (a) retained, (b) modified or (c) replaced (see Tables 3-6).

Score = 46.740738%	pp (np_nm)	-{-}	the {AT0}	weekly {AJ}	meeting {NN}
PP (NP1)					
-{-}		**100%**	0%	0%	0%
one{CRD}		0%	10%	**17%**	0%
charge{NN1}		0%	0%	0%	**30%**
PAD		20%	**20%**	20%	0%

Table 3. Detailed chunk comparison (low similarity).

Score = 48.0%	pp (np_ge)	of {-PRF}	the {AT0}	most {AV0}	boring {AJ}	people {NN}
PP (NP2)						
of{PRF}		**100%**	0%	0%	0%	0%
the{AT0}		0%	**100%**	25%	0%	0%
Battery {NN1}		0%	0%	0%	0%	**30%**
PAD		20%	20%	**20%**	20%	0%
PAD		20%	20%	20%	**20%**	0%

Table 4. Detailed chunk comparison (low similarity).

Score=100.0%	last{VV}
last{VVZ}	**100%**

Table 5. Detailed chunk comparison (high similarity).

Score=100.0%	for{PRP}	hour{NN}
for{PRP}	**100%**	0%
hour{NN2}	0%	**100%**

Table 6. Detailed chunk comparison (high similarity).

Tables 5 and 6 show that the chunks 'last' and 'for hour' are retained and will form part of the output string. The other two chunks (Tables 3 & 4) are handled at the fourth step of the algorithm: the chunk database is searched for appropriate chunk patterns, in an attempt to reduce any incompatibilities between the TL clause pattern and the TL-like one.[5] The chunks that match best with the chunk patterns in the TL-like input string are located and, if necessary, are minimally modified on the basis of co-occurrence information induced from the corpus with statistical means and form part of the output string. If no matching chunks are found, the system indicates the problem, processes the corresponding portion of the TL-like string with co-occurrence information and returns the result.

As explained earlier in this section, the output of the procedure described consists of a sequence of lemmas. Token generation is foreseen for next versions of the system.

4. Evaluation

The system presented has been successfully evaluated for four language pairs (Greek, Spanish, Dutch, German → English) over a test corpus of 60 sentences and compared to the performance of a commercial translation system, namely SYSTRAN.

To that end, widely used benchmarks such as BLEU (Papineni et al. 2002) and NIST (2002) have been employed, which both rely on n-grams of words and adopt a metric that compares experimentally-derived translations to a set of reference translations.

The evaluation results indicated for all four language pairs, the proposed system generated consistently more accurate translations than SYSTRAN, while for some pairs this improvement in accuracy is statistically significant (see Figure 1).

For a more detailed description of the results obtained see Tambouratzis et al. (2006).

[5] Due to space limitations it is not possible to present the whole process in full detail.

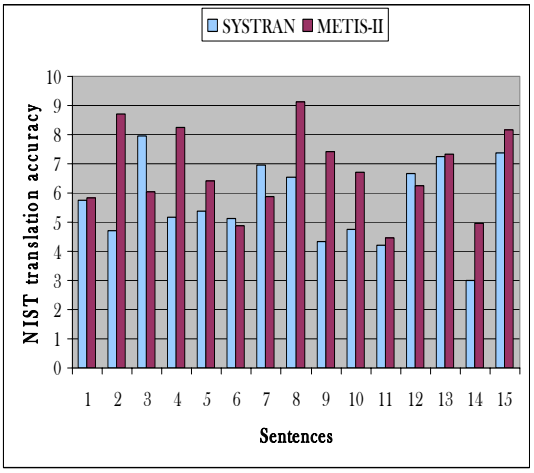

Figure 1: NIST-derived translation accuracies for each of the 15 sentences within the Greek-to-English experiments, for SYSTRAN and the proposed system.

5. Conclusion & further Research

We have reported on the development of a hybrid MT system that relies on monolingual TL corpora, as opposed to all other contemporary corpus-based approaches to MT that rely on parallel corpora. The system employs flat bilingual lexica as well as lemma and chunking information to create TL-patterns which receive their final form (that is, lemmatized grammatical TL strings) by consulting the (chunked and lemmatised) TL corpus with pattern matching techniques.

Pattern matching conceptually relies on a predicate – argument correspondence of the source and target language constructions. This same mechanism handles any categorical (at phrase level) and word-order divergences across the language pair. This set-up captures a large percentage of cases. Of course, there are divergences that can not be captured with this mechanism only, such as the pair 'ανέβηκε την σκάλα τρέχοντας' (EL) → 'he ran up the stairs' (EN) where the SL verb corresponds to a TL particle, while the TL verb corresponds to a SL gerund. However, the work presented here has not fully exploited the potential of the system as no rules have been employed yet and the lexicon contains only one-word entries (and no multi-word entries).

Research in the immediate future will investigate such issues as well as the optimal way of distributing work among the basic pattern matching algorithm, the lexicon and the rule component. In any case, the latter will be kept as small as possible.

6. References

AL-ONAIZAN, Yaser, GERMANN, Ulrich., HERMJAKOB, Ulf, KNIGHT, Kevin, KOEHN, Philipp, MARCU, Daniel, YAMADA, Kenji (2000). Translating with Scarce Resources. American Association for Artificial Intelligence conference (AAAI '00), Austin, Texas, 672-678. Retrieved from www.isi.edu/natural-language/projects/rewrite

BROWN, Peter, COCKE, John, DELLA PIETRA Stephan, DELLA PIETRA Vincent, JELINEK Fredrick, LAFFERTY John, MERCER Robert, ROOSIN Paul (1990). A Statistical Approach to Machine Translation. Computational Linguistics, Vol. 16, No. 2, 79-85.

BROWN, Ralf (2003) Clustered Transfer Rule Induction for Example-Based Translation. In M. Carl & A. Way (eds.) Recent Advances in Example-Based Machine Translation, Kluwer Academic Publishers 287-305.

CARL, Michael & WAY, Andy (2003). Introduction. In M. Carl & A. Way (eds.) Recent Advances in Example-Based Machine Translation. Kluwer Academic Publishers, xvii-xxxi.

DOLOGLOU, Ioannis, MARKANTONATOU, Stella, TAMBOURATZIS, George, YANNOUTSOU, Olga, FOURLA, Athanassia, and IOANNOU, Nikos (2003). 'Using Monolingual Corpora for Statistical Machine Translation'. In Proceedings of EAMT/CLAW 2003, Dublin, Ireland, 61-68.

GAIZAUSKAS, Robert. (1995). Investigations into the Grammar Underlying the Penn Treebank II. Research Memorandum CS-95-25, Department of Computer Science, University of Sheffield.

HUTCHINS, John (1995). Machine Translation: A brief history. In E.F.K. Koerner and R.E. Asher (eds.). Concise history of the language sciences: from the Sumerians to Cognitivists. Oxford: Pergamon Press 431-445.

KITAMURA, Mihoko. (2004). Translation Knowledge Acquisition for Pattern-Based Machine Translation. PhD, Nara Institute of Science and Technology, Japan.

LEPAGE, Yves. (1997). String approximate pattern-matching. In Proceedings of the 55th Meeting of the Information Processing Society of Japan, Fukuoka, August 1997 139-140.

MARKANTONATOU, Stella, SOFIANOPOULOS Sokratis, SPILIOTI Vassiliki, TAMBOURATZIS George,

VASSILIOU Marina, YANNOUTSOU Olga, and Ioannou Nikos (2005). "Monolingual Corpus-based MT using Chunks". In Proceedings of Workshop 'Example Based Machine Translation', 10th MT Summit, September 12-16, Phuket, Thailand 91-98.

MCTAIT, Kevin. (2003). Translation Patterns, Linguistic Knowledge and Complexity in EBMT. In M. Carl & A. Way (eds.): Recent Advances in Example-Based Machine Translation, Kluwer Academic Publishers 307-338.

NAGAO, Makoto (1984). A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. In A. Elithorn and R. Banerji (eds.) Artificial and Human Intelligence, North-Holland.

NEY, Herman. (2005). One Decade of Statistical Machine Translation: 1996-2005. In Proceedings of the 10th MT Summit, September 12-16, Phuket, Thailand, i12-i17.

NIRENBURG, Sergei & RASKIN, Victor (2004). Ontological Semantics. The MIT press.

NIST (2002). Automatic Evaluation of Machine Translation Quality Using n-gram Co-occurrences Statistics. Retrieved from www.nist.gov/speech/tests/mt/

PAPINENI, Kishore, ROUKOS, Salim, WARD, Todd, ZHU, Wei-Jing (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, U.S.A., 311-318.

POPOWICH, Fred, NEY, Herman. (2005). Exploiting Phrasal Lexica and Additional Morpho-Syntactic Language Resources for Statistical Machine Translation with Scarce Training Data. EAMT 10th Annual Conference, Budapest, Hungary.

TAMBOURATZIS George, SOFIANOPOULOS Sokratis, SPILIOTI Vassiliki, VASSILIOU Marina, YANNOUTSOU Olga, and MARKANTONATOU Stella (2006). Pattern matching-based system for Machine Translation (MT). In Proceedings of "Advances in Artificial Intelligence: 4th Hellenic Conference on AI, SETN 2006 (Heraklion, Crete, Greece, May 18-20, 2006), Lecture Notes in Computer Science, Vol. 3955, pp. 345-355. Springer Verlag.

THURMAIR, Gregor. (2005). Improving MT Quality: Towards a Hybrid MT Architecture in the Linguatec 'Personal Translator'. Talk given at the 10th MT Summit, September 12-16, Phuket, Thailand.

WAY, Andy (2003). Translating with Examples: The LFG-DOT Models of Translation, In Recent Advances in Example-Based Machine Translation, Michael Carl and Andy Way (eds.), Kluwer Academic Publishers 443-472.

---

* Author names are given in alphabetical order.