Pattern Recognition Letters 53 (2015) 44–52

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Conditional random fields versus template-matching in MT phrasing tasks involving sparse training data[☆]

George Tambouratzis[*]

Institute for Language and Speech Processing, Athena R.C., 6 Artemidos & Epidavrou Str., Paradissos Amaroussiou, 15125 Athens, Greece

ARTICLE INFO

Article history:
Received 19 November 2013
Available online 27 November 2014

Keywords:
Parsing of natural language
Template-matching
Conditional-random fields
Phrasing model generator
Machine translation

ABSTRACT

This communication focuses on comparing the template-matching technique to established probabilistic approaches – such as conditional random fields (CRF) – on a specific linguistic task, namely the phrasing of a sequence of words into phrases. This task represents a low-level parsing of the sequence into linguistically-motivated phrases. CRF represents the established method for implementing such a data-driven parser, while template-matching is a simpler method that is faster to train and operate. The two aforementioned techniques are compared here to determine the most suitable approach for extracting an accurate model.

The specific application studied is related to a machine translation (MT) methodology (namely PRESEMT), though the comparison performed holds for other applications as well, for which only sparse training data are available. PRESEMT uses small parallel corpora to learn structural transformations from a source language (SL) to a target language (TL) and thus translate input text. This results in the availability of only sparse training data from which to train the parser. Experimental results indicate that for a limited-size training set, as is the case for the PRESEMT methodology, template-matching generates a superior phrasing model that in turn generates higher quality translations. This is confirmed by studying more than one source/target language pairs, for multiple independent testsets.

1. Introduction

Most state-of-the-art MT paradigms translate sentences by operating on parallel corpora at a sub-sentential level using phrases (linguistically-motivated chunks of words). However, the use of phrases in MT naturally assumes the existence of parsers for both SL and TL which develop matched segmentations that either (i) give similar phrasings over the SL and TL or (ii) for which a mapping is defined between the two given segmentations. Both alternatives limit portability to new languages, due to the need for matching the SL and TL parsers to each other, a process which frequently involves a major effort. Another limitation involves the amount of parallel texts needed. In several applications, including the most popular MT paradigm (statistical MT [SMT], [7]), high quality translations require the availability of substantial parallel corpora (containing millions of words).

As a rule, probabilistic parsing approaches trained with large volumes of annotated data (sequences of tokens with tag information, where the boundaries of phrases are marked) are used to create a parser. As reviewed by Collins [2], probabilistic parsers represent a major portion of the highest-performing parsers for natural language processing. In the family of probabilistic parsers, a number of attempts to further improve performance have been reported, including different annotation types [6], hierarchical models for pruning [14] and discriminative methods re-ranking the top n solutions in order to determine the optimal chunking [3].

Within the context of the present article, one critical constraint concerns the volume of available training data. In the present context the parser is developed for use in the PRESEMT methodology (www.presemt.eu), which supports rapid development of MT systems for new language pairs, using pattern recognition principles. PRESEMT utilises a very small parallel corpus of a few hundred sentences, together with a large TL monolingual corpus from which a language model is extracted. PRESEMT analyses these corpora into sub-sentential segments, defined via a TL parser which can be selected by the user. The present work aims at extrapolating the best possible SL parser from the sparse training data of the parallel corpus.

Since the SL and TL phrasing schemes need to be matched, the work reported here relates closely to cross-language approaches transferring phrasing schemes from one language to another, to supplement the sparse data available. Several studies involving the transfer of phrasing schemes across languages have focussed on extrapolating information from a resource-rich to a resource-poor language. For

[☆] This paper has been recommended for acceptance by Y. Chang.

[*] Tel.: +30 210 6875411; fax: +30 210 6854270.
E-mail address: giorg_t@ilsp.gr

http://dx.doi.org/10.1016/j.patrec.2014.11.002
0167-8655/© 2014 Elsevier B.V. All rights reserved.

instance, Yarowski and Ngai [25] use automatically word-aligned raw bilingual corpora to project annotations. Och and Ney [12] use a two-stage process via a dynamic programming algorithm. In contrast, Simard et al. [16] propose a translation method using non-contiguous phrases, to cover additional linguistic phenomena. Hwa et al. [5] have created a parser for a new language based on a set of parallel sentences together with a parser in a frequently-used language, transferring deeper syntactic structure and introducing fix-up rules. Smith et al. [17] create a TL dependency parser by using bilingual text and automatically-derived word alignments.

In comparison to these methods, here the SL parser is generated via a data-driven approach, avoiding any externally-provided fix-up rules, instead extrapolating these as required from the data. Thus, the present article investigates the creation of an efficient parser using sparse resources, but which need not be tied to a specific MT methodology. PRESEMT is referred to here as the platform into which the proposed parsing scheme is integrated and using which the parser effectiveness is judged through the evaluation of the final translation quality.

2. Principles of the PRESEMT paradigm

The PRESEMT translation process is divided into two main phases. The first phase defines the structure of the translated sentence in terms of phrases while the second phase orders tokens within each phrase and implements the disambiguation of candidate translations. PRESEMT employs a two-step approach for splitting arbitrary input text into phrases. To prepare the translation process, word and phrase alignment is performed on a small set of parallel sentences, followed by the extrapolation of a model that segments the SL text. The parsing module presented here is used in the translation preparation phase to define the structure of the input text. Segmentation is limited to identifying the boundaries of the constituent phrases and their types, without implementing a detailed syntactic analysis. To preserve easy portability to new language pairs for PRESEMT, a parser is assumed in only one language (in TL), which pre-processes this side of the parallel corpus. This pre-processing is used to learn SL–TL phrasal mappings, by grouping together corresponding relevant words in order to create matching sub-sentential segments and finally produce a parsing scheme for the SL-side. The processing of a bilingual corpus and the elicitation of the corresponding phrasal information in PRESEMT involves two modules (cf. [18]):

(i) The phrase aligner module (PAM), which performs text alignment at word and phrase level within the parallel corpus. This language-independent method identifies corresponding terms within a language pair, and aligns the words between the two languages, while at the same time creating phrases for the non-parsed side (SL) of the corpus [20].

(ii) The phrasing model generator (PMG), which establishes a phrasing model from the processed parallel corpus. PMG is trained on the SL-side phrases produced by PAM to generate a suitable phrasing model. This model is then employed to segment user-specified text, providing the input to the PRESEMT translation engine, in the form of a sequence of SL-side phrases. The optimisation of this latter process is the topic of the present article.

3. Basic functionality and design of phrasing model generator

The default PMG implementation, as reported in [22] utilises the CRF stochastic model [8,24]. CRF possesses a powerful representation capability and is widely regarded as the model of choice for modelling tasks that need to take into account the environment of a phrase (i.e. neighbouring phrases in the content of a sentence). For the purposes of PRESEMT, CRF was chosen following a number of comparisons.

Initially a rule-based baseline system for parsing SL texts (for Greek, in the specific series of experiments) was developed. The rules, exemplifying the potential inner structure of phrases, were manually created by language specialists and subsequently refined by inspecting the results obtained. A total of three refining iterations were implemented, adding new rules to the model as well as fine-tuning the existing rules. The highest accuracy achieved by this rule-based system reached 75.9%, using a total of 9 phrasing rules, when counting the number of words assigned correct phrase labels. In comparison, a CRF-based model reached 90.0% when trained on the same data, the phrasing error being reduced by 60% over the rule-based system (from 24% to 10%). Another parsing candidate has been Hidden Markov Models (HMM). Using the same data, the best HMM accuracy over different configurations was 81.4%. Hence, in agreement to what has been reported in literature, CRF is substantially more effective, almost halving the error rate in comparison to HMM.

CRF has been widely used for creating parsers (for instance [15,23]). However, due to the expressiveness of the underlying mathematical modelling, CRF requires a large volume of training patterns to extract an accurate model. The disadvantage is that the use of a large parallel corpus compromises portability to new language pairs. Taking into account the PRESEMT constraints, the set of training patterns is a limited-size corpus of only 200 parallel sentences [22]. Even moving from tokens to lemmas and then to part-of-speech tags to reduce the pattern space, it is hard to model accurately all possible phrase types via CRF (in particular for rarer PoS tags).

Lavergne et al. [9] have employed CRF to create the translation model of a statistical MT system. In this effort, they experiment with various features on which to train the CRF. Their training data is limited in comparison to SMT systems, but still amounts to more than 100,000 sentences on which to train CRF. On the contrary, in the system proposed in the present article, CRF is only applied to implement a pre-processing chunking tool that splits input sentences to phrases, making use of a much smaller corpus that is three orders of magnitude smaller than that of Lavergne et al. [9]. Since the training corpus is so much smaller, only a limited set of features are used here (PoS type and case), to train the CRF model. This is in contrast to the approach by Lavergne et al. [9], who assign to the CRF a more central role in the machine translation phase, both splitting the text into groups of tokens and providing re-ordering information, by using richer features (including tokens) together with a larger corpus to implement this functionality.

Within this frame, the template-matching approach, (referred to as TEM, which stands for TEmplate-Matching) has been developed as a relatively naïve system that segments into linguistically-motivated phrases each sentence to be translated. Based on the training data, TEM creates a look-up table of phrases, where for each distinct phrase pattern (determined by the TL-side parser) the length in tokens and the frequency of occurrence are calculated. For the purposes of the present article, the Greek-to-English translation pair is used, according to which the types of phrases (whose tags originate from the TL-side, i.e. English) and the tags of tokens for SL (Greek) are listed in Tables 1 and 2.

Table 1
Main types of phrases for the Greek-to-English language pair (inherited from the TL-side, in this case the TreeTagger parser).

Type	Name	Example
ADVC	Adverbial chunk	“νωρίτερα” [earlier]
ADJC	Adjectival chunk	“ταχύς” [fast]
PC	Prepositional chunk	“Για τη χώρα” [for the country]
VC	Verb chunk	“θα τρέχει” [will run]