HLT-NAACL 2003 Workshop: Building and Using Parallel Texts
Data Driven Machine Translation and Beyond , pp. 19-22
Edmonton, May-June 2003

Statistical Translation Alignment with Compositionality Constraints

Michel Simard and **Philippe Langlais**
Laboratoire de recherche appliquée en linguistique informatique (RALI)
Département d'informatique et de recherche opérationnelle
Université de Montréal
C.P. 6128, succursale Centre-ville, Local 2241
Montréal (Québec), Canada H3C 3J7
{simardm,felipe}@iro.umontreal.ca

Abstract

This article presents a method for aligning words between translations, that imposes a compositionality constraint on alignments produced with statistical translation models. Experiments conducted within the WPT-03 shared task on word alignment demonstrate the effectiveness of the proposed approach.

1 Introduction

Since the pioneering work of the IBM machine translation team almost 15 years ago (Brown et al., 1990), statistical methods have proven to be valuable tools in approaching the automation of translation. Word alignments (WA) play a central role in the statistical modeling process, and reliable WA techniques are crucial in acquiring the parameters of the models (Och and Ney, 2000). Yet, the very nature of these alignments, as defined in the IBM modeling approach (Brown et al., 1993), lead to descriptions of the correspondences between source-language (SL) and target-language (TL) words of a translation that are often unsatisfactory, at least from a human perspective.

One notion that is typically evacuated in the statistical modeling process is that of *compositionality*: a fundamental assumption in statistical machine translation is that, ultimately, *all* the words of a SL segment S contribute to produce *all* the words of its TL translation T, at least to some degree. While this makes perfect sense from a stochastic point of view, it contrasts with the hypothesis at the basis of most (if not all) other MT approaches, as well as with our natural intuitions about translation: that individual portions of the SL text produce individual TL portions autonomously, and that the final translation T is obtained by somehow piecing together these TL portions.

In what follows, we show how re-integrating compositionality into the statistical translation word alignment process leads to better alignments. We first take a closer look at the "standard" statistical WA techniques in section 2, and then propose a way of imposing a compositionality constraint on these techniques in section 3. In section 4, we discuss various implementation issues, and finally present the experimental results of this approach on the WPT-03 shared task on WA in section 5.

2 Statistical Word Alignment

Brown et al. (1993) define a word alignment as a vector $a = a_1...a_m$ that connects each word of a source-language text $S = s_1...s_m$ to a target-language word in its translation $T = t_1...t_n$, with the interpretation that word t_{a_j} is the translation of word s_j in S ($a_j = 0$ is used to denote words of s that do not produce anything in T).

The *Viterbi alignment* between source and target sentences S and T is defined as the alignment $\hat{a}$ whose probability is maximal under some translation model:

$$\hat{a} = \mathrm{argmax}_{a \in \mathcal{A}} \mathrm{Pr}_{\mathcal{M}}(a|S,T)$$

where $\mathcal{A}$ is the set of all possible alignments between S and T, and $\mathrm{Pr}_{\mathcal{M}}(a|S,T)$ is the estimate of a's probability under model $\mathcal{M}$, which we denote $\mathrm{Pr}(a|S,T)$ from hereon. In general, the size of $\mathcal{A}$ grows exponentially with the sizes of S and T, and so there is no efficient way of computing $\hat{a}$ efficiently. However, under the independence hypotheses of IBM Model 2, the Viterbi alignment can be obtained by simply picking for each position i in S, the alignment that maximizes $t(s_i|t_j)a(j,i,m,n)$, the product of the model's "lexical" and "alignment" probability estimates. This procedure can trivially be carried out in $\mathcal{O}(mn)$ operations. Because of this convenient property, we take the Viterbi-2 WA method (which we later refer to as the *V* method) as the basis for the rest of this work.

3 Compositionality

In IBM-style alignments, each SL token is connected to a single (possibly null) TL token, typically the TL token with which it has the most "lexical affinities", regardless of other existing connections in the alignment and, more importantly, of the relationships it holds with other SL tokens in its vicinity. In practice, this means that some TL tokens can end up being connected to several SL tokens, while other TL tokens are left unconnected. This contrasts with alternative alignment models such as those of Melamed (1998) and Wu (1997), which impose a "one-to-one" constraint on alignments. Such a constraint evokes the notion of compositionality in translation: it suggests that each SL token operates independently in the SL sentence to produce a single TL token in the TL sentence, which then depends on no other SL token.

This view is, of course, extreme, and real-life translations are full of examples that show how this compositionality principle breaks down as we approach the level of word correspondences. Yet, if we can find a way of imposing compositionality constraints on WA's, at least to the level where it applies, then we should obtain more sensible results than with Viterbi alignments.

For instance, consider a procedure that splits both the SL and TL sentences S and T into two independent parts, in such a way as to maximise the probability of the two resulting Viterbi alignments:

$$\mathrm{argmax}_{\langle i,j,d\rangle} \left\{ \begin{array}{lll} d = 1 & : & \Pr(a_1|s_1^i, t_1^j) \\ & & \times\Pr(a_2|s_{i+1}^m, t_{j+1}^n) \\ d = -1 & : & \Pr(a_1|s_1^i, t_{j+1}^n) \\ & & \times\Pr(a_2|s_{i+1}^m, t_1^j) \end{array} \right. \tag{1}$$

In the triple $\langle i,j,d\rangle$ above, i represents a "split point" in the SL sentence S, j is the analog for TL sentence T, and d is the "direction of correspondence": $d = 1$ denotes a "parallel correspondence", i.e. $s_1...s_i$ corresponds to $t_1...t_j$ and $s_{i+1}...s_m$ corresponds to $t_{j+1}...t_n$; $d = -1$ denotes a "crossing correspondence", i.e. $s_1...s_i$ corresponds to $t_{j+1}...t_n$ and $s_{i+1}...s_m$ corresponds to $t_1...t_j$.

The triple $\langle I,J,D\rangle$ produced by this procedure refers to the most probable alignment between S and T, under the hypothesis that both sentences are made up of two independent parts ($s_1...s_I$ and $s_{I+1}...s_m$ on the one hand, $t_1...t_J$ and $t_{J+1}...t_n$ on the other), that correspond to each other two-by-two, following direction D. Such an alignment suggests that translation T was obtained by "composing" the translation of $s_1...s_I$ with that of $s_{I+1}...s_m$.

In the above procedure, these "composing parts" of S and T are further assumed to be contiguous subsequences of words. Once again, real-life translations are full of examples that contradict this (negations in French and particle verbs in German are two examples that immediately spring to mind when aligning with English). Yet, this *contiguity assumption* turns out to be very convenient, because examining pairings of non-contiguous sequences would quickly become intractable. In contrast, the procedure above can find the optimal partition in polynomial time.

The "splitting" process described above can be repeated recursively on each pair of matching segments, down to the point where the SL segment contains a single token. (TL segments can always be split, even when empty, because IBM-style alignments allow connecting SL tokens to the "null" TL token, which is always available.) This recursive procedure actually produces two different outputs:

1. A parallel partition of S and T into m pairs of segments $\langle s_i, t_j^k\rangle$, where each t_j^k is a (possibly null) contiguous sub-sequence of T; this partition can of course be viewed as an alignment on the words of S and T.

2. an IBM-style alignment, such that each SL and TL token is linked to at most one token in the other language: this alignment is actually the concatenation of individual Viterbi alignments on the $\langle s_i, t_j^k\rangle$ pairs, which connects each s_i to (at most) one of the tokens in the corresponding t_j^k.

In this procedure, which we call *Compositional WA* (or *C* for short), there are at least two problems. First, each SL token finds itself "isolated" in its own partition bin, which makes it impossible to account for multiple SL tokens acting together to produce a TL sequence. Second, the TL tokens that are not connected in the resulting IBM-style alignment do not play any role in the computation of the probability of the optimal alignment; therefore, the pair $\langle s_i, t_j^k\rangle$ in which these "superfluous" tokens end up is more or less random.

To compensate in part for these, we propose using two IBM-2 models to compute the optimal partition: the "forward" (SL→TL) model, and the "reverse" (TL→SL) model. When examining a particular split $\langle i,j,d\rangle$ for S and T, we compute both Viterbi alignments, forward and reverse, between all pairs of segments, and score each pair with the product of the two alignments' probabilities.

In this variant, which we call *Combined Compositional WA* (*CC*), we can no longer allow "empty" segments in the TL, and so we stop the recursion as soon as either the SL or TL segment contains a single token. The resulting partition therefore consists in a series of 1-to-k or k-to-1 alignments, with $k \geq 1$.

4 Implementation

The *C* and *CC* WA methods of section 3 were implemented in a program called *ralign* (*R*ecursive – or *RALI* – alignment, as you wish). As suggested above, this program takes as input a pair of sentence-aligned texts, and the parameters of two IBM-2 models (forward and reverse), and outputs WA's for the given texts. This program also implements plain Viterbi alignments, using the forward (*V*) or reverse (*RV*) models, as well as what we call the *Reverse compositional WA* (or *RC*), which is just the *C* method using the reverse IBM-2 model.

The output format proposed for the WPT-03 shared task on WA allowed participants to distinguish between "sure" (S) and "probable" (P) WA's. We figured that our alignment procedure implicitly incorporated a way of distinguishing between the two: within each produced pair of segments, we marked as "sure" all WA's that were predicted by both (forward and reverse) Viterbi alignments, and as "probable" all the others.

The translation models for *ralign* were trained using the programs of the *EGYPT* statistical translation toolkit (Al-Onaizan et al., 1999). This training was done using the data provided as part of the WPT-03 shared task on WA (table 1). We thus produced two sets of models, one for English and French (*en-fr*), and one for Romanian and English (*ro-en*). All models were trained on both the *training* and *test* datasets[1]. For *en-fr*, we considered all words that appeared only once in the corpus to be "unknown words" (*whittle* option `-f 2`), so as to obtain default values of "real" unknowns in the test corpus[2]. In the case of *ro-en*, there was too little training data for this to be beneficial, and so we chose to use all words.

English-French

corpus	tokens (SL/TL)	sentence pairs
training	20M/24M	1M
trial	772/832	37
test	8K/9K	447

Romanian-English

corpus	tokens (SL/TL)	sentence pairs
training	1M/1M	48K
trial	513/547	17
test	6K/6K	248

Table 1: WPT-03 shared task resources

We trained and tested a number of translation models before settling for this particular setup. All of these tests were performed using the *trial* data provided for the WPT-03 shared task.

5 Experimental Results

The different word-alignment methods described in sections 2 and 3 were run on the test corpora of the WPT-03 shared task on alignment. Results were evaluated in terms of alignment precision (P), recall (R), F-measure and *alignment error rate* (AER) (Och and Ney, 2000). As specified in the shared task description, all of these metrics were computed taking *null*-alignments into account (i.e. tokens left unconnected in an alignment were actually counted as aligned to virtual word token "0"). The results of our experiments are reproduced in table 2.

We observe that imposing a "contiguous compositionality" constraint (*C* and *RC* methods) allows for substantial gains with regard to plain Viterbi alignments (*V* and *RV* respectively), especially in terms of precision and AER (a slight decline in recall can be observed between the *V* and *C* methods on the *ro-en* corpus, but it is not clear whether this is significant). These gains are even more interesting when one considers that all pairs of alignments (*V* and *C*, *RV* and *RC*) are obtained using exactly the same data. This highlights both the deficiencies of IBM Model-2 and the importance of compositionality.

Using both the forward and reverse models (*CC*) yields yet more gains with regard to all metrics. This result is interesting, because it shows the potential of the compositional alignment method for integrating various sources of information.

With regard to language pairs, it is interesting to note that all alignment methods produce figures that are substantially better in recall and worse in precision on the *ro-en* data, compared to *en-fr*. Overall, *ro-en* alignments display significantly higher F-measures. This is surprising, considering that the provided *en-fr* corpus contained 20 times more training material. This phenomenon is likely due to the fact that the *en-fr* test reference contains much more alignments per word (1.98 per target word) than the *ro-en* (1.12). All alignment methods described here produce roughly between 1 and 1.25 alignments per target words. This fact affects recall and F-measure figures positively on the *ro-en* test, while precision and AER (which correlates strongly with precision in practice) are affected inversely.

6 Conclusion

In this article, we showed how a compositionality constraint could be imposed when computing word alignments with IBM Models-2. Our experiments on the WPT-03 shared task on WA demonstrated how this improves the quality of resulting alignments, when compared to standard Viterbi alignments. Our results also highlight

[1] No cheating here: the *test* dataset did not contain reference alignments

[2] This is necessary, even when training on the test corpus, because the EGYPT toolkit's training program (GIZA) ignores excessively long sentences in the corpus.

English-French				
method	P	R	F	AER
V	0.6610	0.3387	0.4479	0.2700
RV	0.6260	0.3212	0.4245	0.2944
C	0.7248	0.3534	0.4751	0.2318
RC	0.7422	0.3586	0.4835	0.2152
CC	0.7756	0.3681	0.4992	0.1850

Romanian-English				
method	P	R	F	AER
V	0.5509	0.5442	0.5475	0.4524
RV	0.5409	0.5375	0.5391	0.4608
C	0.5818	0.5394	0.5597	0.4402
RC	0.5865	0.5415	0.5630	0.4369
CC	0.6361	0.5714	0.6020	0.3980

Table 2: Alignment results

the benefit of using both forward and reverse translation models for this task.

One of the weaknesses of the proposed method is the inability to produce many-to-many alignments. To allow for such alignments, it would be necessary to establish a “stopping condition” on the recursion process, so as to prevent partitioning pairs of segments that display “non-compositional” phenomena in both SL and TL languages. We have begun experimenting with various such mechanisms. One of these is to stop the recursion as soon as the pair of segments under consideration contains less than two “sure” alignments, i.e. connections predicted by both the forward and reverse models. Another possibility is to establish a threshold on the probability “drop” incurred by the optimal split on any given pair of segments. So far, these experiments are inconclusive.

Another problem is with “null” alignments, which the program is also unable to account for. Currently, omissions and insertions in translation find themselves incorporated into aligned segments. A simple way to deal with this problem would be to exclude from the final alignment links that are not predicted by either the forward or reverse Viterbi alignments. But early experiments with this approach are unconvincing, and more elaborate filtering mechanisms will probably be necessary.

Finally, IBM Model 2 is certainly not the state of the art in statistical translation modeling. Thenagain, the methods proposed here are not dependent on the underlying translation model, and similar WA methods could be based on more elaborate models, such as Models 3–5, or the HMM-based models proposed by Och et al. (1999) for example. On the other hand, our compositional alignment method could be used during the training process of higher-level models. Whether this would lead to better estimates of the models’ parameters remains to be seen, but it is certainly a direction worth exploring.

References

[Al-Onaizan et al.1999] Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz-Josef Och, David Purdy, Noah H. Smith, and David Yarowsky. 1999. Statistical Machine Translation - Final Report, JHU Workshop 1999. Technical report, Johns Hopkins University.

[Brown et al.1990] Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics*, 16(2):79–85, June.

[Brown et al.1993] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

[Melamed1998] I. Dan Melamed. 1998. Word-to-Word Models of Translational Equivalence. Technical Report 98-08, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, USA.

[Och and Ney2000] Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 440–447, Hong-Kong, China, October.

[Och et al.1999] Franz Josef Och, Christoph Tillmann, and Hermann Ney. 1999. Improved Alignment Models for Statistical Machine Translation. In *Proceedings of the 4th Conference on Empirical Methods in Natural Language Processing (EMNLP)and 7th ACL Workshop on Very Large Corpora (WVLC)*, pages 20–28, College Park, USA.

[Wu1997] Dekai Wu. 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Linguistics*, 23(3):377–404, September.