Improved HMM Alignment Models for Languages with Scarce Resources

Adam Lopez
Institute for Advanced Computer Studies
Department of Computer Science
University of Maryland
College Park, MD 20742
alopez@cs.umd.edu

Philip Resnik
Institute for Advanced Computer Studies
Department of Linguistics
University of Maryland
College Park, MD 20742
resnik@umiacs.umd.edu

Abstract

We introduce improvements to statistical word alignment based on the Hidden Markov Model. One improvement incorporates syntactic knowledge. Results on the workshop data show that alignment performance exceeds that of a state-of-the art system based on more complex models, resulting in over a 5.5% absolute reduction in error on Romanian-English.

1 Introduction

The most widely used alignment model is IBM Model 4 (Brown et al., 1993). In empirical evaluations it has outperformed the other IBM Models and a Hidden Markov Model (HMM) (Och and Ney, 2003). It was the basis for a system that performed very well in a comparison of several alignment systems (Dejean et al., 2003; Mihalcea and Pedersen, 2003). Implementations are also freely available (Al-Onaizan et al., 1999; Och and Ney, 2003).

The IBM Model 4 search space cannot be efficiently enumerated; therefore it cannot be trained directly using Expectation Maximization (EM). In practice, a sequence of simpler models such as IBM Model 1 and an HMM Model are used to generate initial parameter estimates and to enumerate a partial search space which can be expanded using hill-climbing heuristics. IBM Model 4 parameters are then estimated over this partial search space as an approximation to EM (Brown et al., 1993; Och and Ney, 2003). This approach yields good results, but it has been observed that the IBM Model 4 performance is only slightly better than that of the underlying HMM Model used in this bootstrapping process (Och and Ney, 2003). This is illustrated in Figure 1.

Based on this observation, we hypothesize that implementations of IBM Model 4 derive most of their performance benefits from the underlying HMM Model. Furthermore, owing to the simplicity of HMM Models, we believe that they are more conducive to study and improvement than more complex models such as IBM Model 4. We illustrate this point by introducing modifications to the HMM model which improve performance.

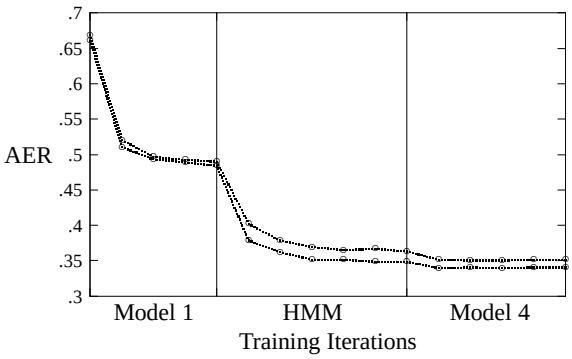


Figure 1: The improvement in Alignment Error Rate (AER) is shown for both $P(\mathbf{f}|\mathbf{e})$ and $P(\mathbf{e}|\mathbf{f})$ alignments on the Romanian-English development set over several iterations of the IBM Model 1 → HMM → IBM Model 4 training sequence.

2 HMMs and Word Alignment

The objective of word alignment is to discover the word-to-word translational correspondences in a bilingual corpus of S sentence pairs, which we denote $\{(\mathbf{f}^{(s)}, \mathbf{e}^{(s)}) : s \in [1,S]\}$. Each sentence pair $(\mathbf{f},\mathbf{e}) = (f_1^M, e_1^N)$ consists of a sentence $\mathbf{f}$ in one language and its translation $\mathbf{e}$ in the other, with lengths M and N, respectively. By convention we refer to $\mathbf{e}$ as the English sentence and $\mathbf{f}$ as the French sentence. Correspondences in a sentence are represented by a set of links between words. A link (f_j, e_i) denotes a correspondence between the ith word e_i of $\mathbf{e}$ and the jth word f_j of $\mathbf{f}$.

Many alignment models arise from the conditional distribution $P(\mathbf{f}|\mathbf{e})$. We can decompose this by introducing the hidden alignment variable $\mathbf{a} = a_1^M$. Each element of $\mathbf{a}$ takes on a value in the range $[1,N]$. The value of a_i determines a link between the ith French word f_i and the a_ith English word e_{a_i}. This representation introduces

Proceedings of the ACL Workshop on Building and Using Parallel Texts, pages 83–86,
Ann Arbor, June 2005.

an asymmetry into the model because it constrains each French word to correspond to exactly one English word, while each English word is permitted to correspond to an arbitrary number of French words. Although the resulting set of links may still be relatively accurate, we can symmetrize by combining it with the set produced by applying the complementary model $P(\mathbf{e}|\mathbf{f})$ to the same data (Och and Ney, 2000b). Making a few independence assumptions we arrive at the decomposition in Equation 1. [1]

$$P(\mathbf{f},\mathbf{a}|\mathbf{e}) = \prod_{i=1}^{M} d(a_i|a_{i-1}) \cdot t(f_i|e_{a_i}) \quad (1)$$

We refer to $d(a_i|a_{i-1})$ as the *distortion model* and $t(f_i|e_{a_i})$ as the *translation model*. Conveniently, Equation 1 is in the form of an HMM, so we can apply standard algorithms for HMM parameter estimation and maximization. This approach was proposed in Vogel et al. (1996) and subsequently improved (Och and Ney, 2000a; Toutanova et al., 2002).

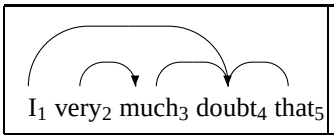
2.1 The Tree Distortion Model

Equation 1 is adequate in practice, but we can improve it. Numerous parameterizations have been proposed for the distortion model. In our *surface distortion* model, it depends only on the distance $a_i - a_{i-1}$ and an automatically determined word class $C(e_{a_{i-1}})$ as shown in Equation 2. It is similar to (Och and Ney, 2000a). The word class $C(e_{a_{i-1}})$ is assigned using an unsupervised approach (Och, 1999).

$$d(a_i|a_{i-1}) = p(a_i|a_i - a_{i-1}, C(e_{a_{i-1}})) \quad (2)$$

The surface distortion model can capture local movement but it cannot capture movement of structures or the behavior of long-distance dependencies across translations. The intuitive appeal of capturing richer information has inspired numerous alignment models (Wu, 1995; Yamada and Knight, 2001; Cherry and Lin, 2003). However, we would like to retain the simplicity and good performance of the HMM Model.

We introduce a distortion model which depends on the *tree distance* $\tau(e_i, e_k) = (w, x, y)$ between each pair of English words e_i and e_k. Given a dependency parse of e_1^M, w and x represent the respective number of dependency links separating e_i and e_k from their closest common ancestor node in the parse tree. [2] The final element $y = \{1$ if $i > k$; 0 otherwise$\}$ is simply a binary indicator of the linear relationship of the words within the surface string. Tree distance is illustrated in Figure 2.



$\tau(\text{I}_1, \text{very}_2) = (1,2,0)$
$\tau(\text{very}_2, \text{I}_1) = (2,1,1)$
$\tau(\text{I}_1, \text{doubt}_4) = (1,0,0)$
$\tau(\text{that}_5, \text{I}_1) = (1,1,1)$

Figure 2: Example of tree distances in a sentence from the Romanian-English development set.

In our *tree distortion* model, we condition on the tree distance and the part of speech $T(e_{i-1})$, giving us Equation 3.

$$d(a_i|a_{i-1}) = p(a_i, |\tau(e_{a_i}, e_{a_{i-1}}), T(e_{a_{i-1}})) \quad (3)$$

Since both the surface distortion and tree distortion models represent $p(a_i|a_{i-1})$, we can combine them using linear interpolation as in Equation 4.

$$\begin{aligned} &d(a_i|a_{i-1}) = \\ &\quad \lambda_{C(e_{a_{i-1}}),T(e_{a_{i-1}})} p(a_i|\tau(e_{a_i}, e_{a_{i-1}}), T(e_{a_{i-1}})) + \\ &\quad (1 - \lambda_{C(e_{a_{i-1}}),T(e_{a_{i-1}})}) p(a_i|a_i - a_{i-1}, C(e_{a_{i-1}})) \end{aligned} \quad (4)$$

The $\lambda_{C,T}$ parameters can be initialized from a uniform distribution and trained with the other parameters using EM. In principle, any number of alternative distortion models could be combined with this framework.

2.2 Improving Initialization

Our HMM produces reasonable results if we draw our initial parameter estimates from a uniform distribution. However, we can do better. We estimate the initial translation probability $t(f_j|e_i)$ from the smoothed log-likelihood ratio $LLR(e_i, f_j)^{\phi_1}$ computed over sentence cooccurrences. Since this method works well, we apply $LLR(e_i, f_j)$ in a single reestimation step shown in Equation 5.

$$t(f|e) = \frac{LLR(f|e)^{\phi_2} + n}{\sum_{e'} LLR(f|e')^{\phi_2} + n \cdot |V|} \quad (5)$$

In reestimation $LLR(f|e)$ is computed from the expected counts of f and e produced by the EM algorithm. This is similar to Moore (2004); as in that work, $|V| = 100,000$, and ϕ_1, ϕ_2, and n are estimated on development data.

We can also use an improved initial estimate for distortion. Consider a simple distortion model $p(a_i|a_i - a_{i-1})$. We expect this distribution to have a maximum near $P(a_i|0)$ because we know that words tend to retain their locality across translation. Rather than wait for this to occur, we use an initial estimate for the distortion model given in Equation 6.

[1] We ignore the sentence length probability $p(M|N)$, which is not relevant to word alignment. We also omit discussion of HMM start and stop probabilities, and normalization of $t(f_i|e_{a_i})$, although we find in practice that attention to these details can be beneficial.

[2] The tree distance could easily be adapted to work with phrase-structure parses or tree-adjoining parses instead of dependency parses.

corpus	n	ϕ_1	ϕ_2	α	symmetrization	n^{-1}	ϕ_1^{-1}	ϕ_2^{-1}	α^{-1}
English-Inuktitut	1^{-4}	1.0	1.75	-1.5	$\cap$	5^{-4}	1.0	1.75	-1.5
Romanian-English	5^{-4}	1.5	1.0	-2.5	refined (Och and Ney, 2000b)	5^{-4}	1.5	1.0	-2.5
English-Hindi	1^{-4}	1.5	3.0	-2.5	$\cup$	1^{-2}	1.0	1.0	-1.0

Table 1: Training parameters for the workshop data (see Section 2.2). Parameters n, ϕ_1, ϕ_2, and α were used in the initialization of $P(\mathbf{f}|\mathbf{e})$ model, while n^{-1}, ϕ_1^{-1}, ϕ_2^{-1}, and α^{-1} were used in the initialization of the $P(\mathbf{e}|\mathbf{f})$ model.

corpus	type	**HMM limited (Eq. 2)**			**HMM unlimited (Eq. 4)**			IBM Model 4		
		P	R	AER	P	R	AER	P	R	AER
English-Inuktitut	P(f|e)	.4962	.6894	.4513	–	–	–	.4211	.6519	.5162
	P(e|f)	.5789	**.8635**	.3856	–	–	–	.5971	.8089	.3749
	$\cap$	**.8916**	.6280	**.2251**	–	–	–	.8682	.5700	.2801
English-Hindi	P(f|e)	.5079	.4769	.5081	.5057	.4748	.5102	.5219	.4223	.5332
	P(e|f)	.5566	.4429	.5067	.5566	.4429	.5067	.5652	.3939	.5358
	$\cup$	.4408	**.5649**	.5084	.4365	.5614	.5088	**.4543**	.5401	**.5065**
Romanian-English	P(f|e)	.6876	**.6233**	.3461	.6876	**.6233**	.3461	.6828	.5414	.3961
	P(e|f)	.7168	.6217	.3341	.7155	.6205	.3354	.7520	.5496	.3649
	refined	.7377	.6169	**.3281**	.7241	.6215	.3311	**.7620**	.5134	.3865

Table 2: Results on the workshop data. The systems highlighted in bold are the ones that were used in the shared task. For each corpus, the last row shown represents the results that were actually submitted. Note that for English-Hindi, our self-reported results in the unlimited task are slightly lower than the original results submitted for the workshop, which contained an error.

$$d(a_i|a_{i-1}) = \begin{cases} |a_i - a_{i-1}|^{\alpha}/Z, \alpha < 0 & \text{if } a_i \neq a_{i-1}. \\ 1/Z & \text{if } a_i = a_{i-1}. \end{cases} \quad (6)$$

We choose Z to normalize the distribution. We must optimize α on a development set. This distribution has a maximum when $|a_i - a_{i-1}| \in \{-1, 0, 1\}$. Although we could reasonably choose any of these three values as the maximum for the initial estimate, we found in development that the maximum of the surface distortion distribution varied with $C(e_{a_i-1})$, although it was always in the range $[-1, 2]$.

2.3 Does NULL Matter in Asymmetric Alignment?

Och and Ney (2000a) introduce a NULL-alignment capability to the HMM alignment model. This allows any word f_j to link to a special NULL word – by convention denoted e_0 – instead of one of the words e_1^N. A link (f_j, e_0) indicates that f_j does not correspond to any word in **e**. This improved alignment performance in the absence of symmetrization, presumably because it allows the model to be conservative when evidence for an alignment is lacking.

We hypothesize that NULL alignment is unnecessary for asymmetric alignment models when we symmetrize using intersection-based methods (Och and Ney, 2000b). The intuition is simple: if we don't permit NULL alignments, then we expect to produce a high-recall, low-precision alignment; the intersection of two such alignments should mainly improve precision, resulting in a high-recall, high-precision alignment. If we allow NULL alignments, we may be able produce a high-precision, low-recall asymmetric alignment, but symmetrization by intersection will not improve recall.

3 Results with the Workshop Data

In our experiments, the dependency parse and parts of speech are produced by minipar (Lin, 1998). This parser has been used in a much different alignment model (Cherry and Lin, 2003). Since we only had parses for English, we did not use tree distortion in the application of $P(\mathbf{e}|\mathbf{f})$, needed for symmetrization.

The parameter settings that we used in aligning the workshop data are presented in Table 1. Although our prior work with English and French indicated that intersection was the best method for symmetrization, we found in development that this varied depending on the characteristics of the corpus and the type of annotation (in particular, whether the annotation set included probable alignments). The results are summarized in Table 2. It shows results with our HMM model using both Equations 2 and 4 as our distortion model, which represent

the unlimited and limited resource tracks, respectively. It also includes a comparison with IBM Model 4, for which we use a training sequence of IBM Model 1 (5 iterations), HMM (6 iterations), and IBM Model 4 (5 iterations). This sequence performed well in an evaluation of the IBM Models (Och and Ney, 2003).

For comparative purposes, we show results of applying both $P(\mathbf{f}|\mathbf{e})$ and $P(\mathbf{e}|\mathbf{f})$ prior to symmetrization, along with results of symmetrization. Comparison of the asymmetric and symmetric results largely supports the hypothesis presented in Section 2.3, as our system generally produces much better recall than IBM Model 4, while offering a competitive precision. Our symmetrized results usually produced higher recall and precision, and lower alignment error rate.

We found that the largest gain in performance came from the improved initialization. The combined distortion model (Equation 4), which provided a small benefit over the surface distortion model (Equation 2) on the development set, performed slightly worse on the test set.

We found that the dependencies on $C(e_{a_{i-1}})$ and $T(e_{a_{i-1}})$ were harmful to the $P(\mathbf{f}|\mathbf{e})$ alignment for Inuktitut, and did not submit results for the unlimited resources configuration. However, we found that alignment was generally difficult for all models on this particular task, perhaps due to the agglutinative nature of Inuktitut.

4 Conclusions

We have proposed improvements to the largely overlooked HMM word alignment model. Our improvements yield good results on the workshop data. We have additionally shown that syntactic information can be incorporated into such a model; although the results are not superior, they are competitive with surface distortion. In future work we expect to explore additional parameterizations of the HMM model, and to perform extrinsic evaluations of the resulting alignments by using them in the parameter estimation of a phrase-based translation model.

Acknowledgements

This research was supported in part by ONR MURI Contract FCPO.810548265. The authors would like to thank Bill Byrne, David Chiang, Okan Kolak, and the anonymous reviewers for their helpful comments.

References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation: Final report. In *Johns Hopkins University 1999 Summer Workshop on Language Engineering*.

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, Jun.

Colin Cherry and Dekang Lin. 2003. A probability model to improve word alignment. In *ACL Proceedings*, Jul.

Herve Dejean, Eric Gaussier, Cyril Goutte, and Kenji Yamada. 2003. Reducing parameter space for word alignment. In *Proceedings of the Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 23–26, May.

Dekang Lin. 1998. Dependency-based evaluation of minipar. In *Proceedings of the Workshop on the Evaluation of Parsing Systems*, May.

Rada Mihalcea and Ted Pedersen. 2003. An evaluation exercise for word alignment. In *Proceedings of the Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10, May.

Robert C. Moore. 2004. Improving IBM word-alignment model 1. In *ACL Proceedings*, pages 519–526, Jul.

Franz Josef Och and Hermann Ney. 2000a. A comparison of alignment models for statistical machine translation. In *COLING Proceedings*, pages 1086–1090, Jul.

Franz Josef Och and Hermann Ney. 2000b. Improved statistical alignment models. In *ACL Proceedings*, pages 440–447, Oct.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison on various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 1999. An efficient method for determining bilingual word classes. In *EACL Proceedings*, pages 71–76, Jun.

Kristina Toutanova, H. Tolga Ilhan, and Christopher D. Manning. 2002. Extensions to hmm-based statistical word alignment models. In *EMNLP*, pages 87–94, Jul.

Stephan Vogel, Hermann Ney, and Christoph Tillman. 1996. Hmm-based word alignment in statistical machine translation. In *COLING Proceedings*, pages 836–841, Aug.

Dekai Wu. 1995. Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1328–1335, Aug.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *ACL Proceedings*.