# Word Association Models and Search Strategies for Discriminative Word Alignment

Patrik Lambert[1] and Rafael E. Banchs[2]

[1] TALP Research Center, Jordi Girona Salgado 1 3, 08034 Barcelona, Spain*
lambert@gps.tsc.upc.edu
[2] Barcelona Media Innovation Centre, Ocata 1, Barcelona 08003, Spain.
rafael.banchs@barcelonamedia.org

**Abstract.** This paper deals with core aspects of discriminative word alignment systems, namely basic word association models as well as search strategies. We compare various low-computational-cost word association models: $\chi^2$ score, log-likelihood ratio and IBM model 1. We also compare three beam-search strategies. We show that it is more flexible and accurate to let links to the same word compete together, than introducing them sequentially in the alignment hypotheses, which is the strategy followed in several systems.

## 1 Introduction

In this paper, we study core aspects of discriminative alignment systems [1, 2]. In these systems, the best alignment hypothesis is the one that maximises a linear combination of features. In Sect. 2 we propose some improvements of the beam-search algorithm implemented by Moore [1]. Then we present experimental results for different low-computational-cost word association score features (Sect. 3.1) and for the proposed search strategies (Sect. 3.2). Finally, we give some conclusions.

## 2 Search Strategies

Search aims at finding the alignment (*i.e.* the set of links between source and target words) which maximises the sum of each feature cost, weighted by its respective weight. In order to limit the search space, a set of promising links is first selected. Then alignment hypotheses are created by introducing some of these promising links, and the cost of each feature function for these alignment hypotheses is calculated.

Figure 1 shows the list of promising links considered (referred to as the list of *possible links*). This list is obtained by pruning the word association feature table[3] with a threshold $N$. Only the best $N$ target words for each source word, *and* the best $N$ source words for each target word are considered. Possible links are arranged in a certain number of stacks of links to be expanded during search.

* This work has been partially funded by the Spanish Government under grant TEC2006-13964-C03 (AVIVAVOZ project).

[3] It contains the word association score for each word pair seen in the training corpus.

**Sentence pair:**
(0)the (1)member (2)state (3).
(0)los (1)pais (2)miembr (3).

**Possible links (word association cost order):**

| Link | Cost | Corresponding words |
|---|---|---|
| 1-2 | 0.1736 | member-miembr |
| 3-3 | 0.6758 | .-. |
| 0-0 | 1.3865 | the-los |
| 2-2 | 1.8285 | state-miembr |
| 2-1 | 2.4027 | state-pais |

**Fig. 1.** Possible links list example. Word position is indicated in parentheses. Corresponding words are actually stemmed forms. Here $N = 1$. Notice that “state” is involved in two links because it is the best source word for both “miembr” and “pais”. The best alignment here would be {0-0,1-2,2-1,3-3}. The cost is $-\log \chi^2$ (see Sect. 3.1).

### 2.1 Baseline Search

With Moore’s search strategy, which will be referred to as *baseline search*, links of the example of Fig. 1 are arranged as depicted in Fig. 2 (left figure). Thus in the baseline search the possible links, sorted in function of their cost, are arranged one link per stack, together with the “empty” link set $\emptyset$. Baseline search always begins with the empty alignment (alignment stack 0).[4] This hypothesis is expanded with each link of link stack 1 forming two new hypotheses (the empty alignment and the alignment containing the link 1-2) which are copied into alignment stack 1. Each hypothesis of alignment stack $i$ is expanded with each link of link stack $i + 1$. Histogram and/or threshold pruning are applied to the alignment hypothesis stack to reduce complexity. The dashed line in alignment stack 2 illustrates the histogram pruning threshold for a beam size of 3.

In our view, the main drawback of the baseline search strategy is that the final alignment depends on the order in which links are introduced. To understand this better, consider a very simple system with a word association feature, a distortion feature and an unlinked word penalty feature. Distortion costs are caused by crossings between links. Each time some unlinked word becomes linked, the unlinked word penalty decreases. When a hypothesis is expanded with a new link, if the word association cost for this link plus a possible distortion cost is smaller than a possible decrease in the unlinked word penalty, the hypothesis with the new link is better than the previous one. In the example of Fig. 2 (left figure), suppose that this was the case successively for links 1-2, 3-3 and 0-0, so that the best alignment hypothesis is {1-2, 3-3, 0-0}. Now if this hypothesis is expanded with link 2-2, the association cost is compensated by the decrease of the unlinked feature cost for “state”, and the new best hypothesis will include link 2-2. Expanding now this last hypothesis with link 2-1, the unlinked feature gain for “pais” cannot compensate for the distortion feature cost (due to crossing

[4] Melamed [3] also starts with the empty alignment and links are added from most to least probable.

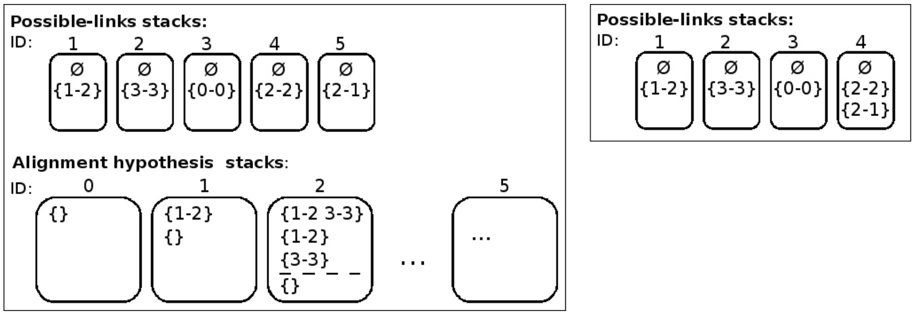

**Fig. 2.** Left: Baseline search: link-by-link search following word association score order [1]. Right: "source-word-score" search strategy.

with "member-miembr") plus the association cost. Thus link 2-1 is not included in the final hypothesis. On the contrary, if we would expand the hypotheses with link 2-1 first, the double unlinked feature gain (for "pais" and "state") would compensate for the other costs, and link 2-1 would appear in the final hypothesis.

Thus in the previous case, a probable but incorrect link (2-2) introduced first prevented the correct link (2-1) from being in the final alignment, because of the unlinked feature. In other situations, this may occur with the distortion feature, the presence of the incorrect link causing a crossing with the correct one. Actually in many cases, when introducing link 2-1, both the new hypothesis (with link 2-2) and the former one (without it) will be in the stack. However, when introducing a link, it can happen that all hypotheses which do not contain a previously introduced link have been pruned out. In this case all hypotheses would contain the link 2-2 when expanding hypotheses with link 2-1, and the problem described above would happen.

### 2.2 Proposed Improvements

To help overcome this problem, we perform successive iterations of the alignment algorithm. In the second one, we start from the final alignment of the first iteration instead of the empty alignment. Expanding a hypothesis with some link still means introducing this link in the alignment hypothesis if it is not present yet, but also means removing it if it is already present. Thus alignment hypotheses now always contain a reasonable set of links for this sentence pair: the first iteration's final links at the start, which are then updated link by link during search. When a hypothesis is expanded with an incorrect link, this link is typically situated (considering the alignment matrix) apart from the rest of links in the hypothesis, causing a distortion cost. If a hypothesis containing no link would be expanded with this incorrect link, it would not be penalised by any distortion cost.

Another idea to alleviate the problem is to let links to the same word compete on a fair basis, considering them at the same time instead of successively in the alignment hypotheses. In this scheme, possible links are organised in one stack for each source (or target) word,[5] as in Fig. 2 (right figure). This is a one-stack-per-word strategy, whereas the baseline search is a one-stack-per-link strategy. The links of each stack are used to expand the same hypotheses. Thus, in our example, expanding hypothesis {1-2, 3-3, 0-0}, 2-1 would have been preferred over 2-2.

In Fig. 2 (right figure), link stacks are sorted according to the cost of the best link in the stack. We will refer to this strategy as "source-word-score" (SWS) search. We could also sort the link stacks according to the source word position, which will be referred to as "source-word-position" (SWP) search.

The total number of alignment hypotheses created during search is the same with both baseline and one-stack-per-word strategies, since the number of "possible links" is equal. However, the one-stack-per-word search, as depicted in Fig. 2, only allows many-to-one links since each hypothesis can only be expanded with one of the various possible links to the same source word. To allow many-to-many links, the stacks of possible links associated with a given word must also contain combinations of these links. Each combination represents an additional alignment hypothesis to create compared to the baseline search. However, the one-stack-per-word strategies also offer more flexibility to control complexity than the baseline strategy. The link stacks can be sorted and pruned by histogram and/or threshold. We can also limit the number of links in the combinations, or allow only combinations with consecutive target positions. One-stack-per-word strategies also make it easy to first expand words with a higher confidence or less ambiguity. This gives a context of links which helps aligning the other words.

Note that an adequate solution to the problem raised in Sect. 2.1 would be to estimate *exactly* the remaining cost of each hypothesis, but this would be too expensive computationally. In one-stack-per-word strategies, the future word association cost (considering the most probable path) is not useful because it would be the same in each stack, since the same words have been covered. We estimated a relative distortion cost of each link with respect to the best links (in terms of word association score) for surrounding words remaining to cover. However, this estimation was too inaccurate and did not improve our results.

## 3 Experiments

We used freely available[6] alignment test data [4]. These data are a subset of the training corpus: the TC-STAR OpenLabSpanish-English EPPS parallel corpus, which contains proceedings of the European Parliament. The training corpus

[5] This is much more efficient than Liu *et al.*'s search [2], which considers *all* possible links before selecting each link.

[6] `http://gps-tsc.upc.es/veu/LR`

contains 1.28 million sentence pairs of respectively 27.2 and 28.5 words length in average for English and Spanish. English and Spanish vocabulary size are respectively 106 and 153 thousand words. We divided randomly the alignment reference corpus in a 246-sentence development set and a 245-sentence test set.

Evaluation was done with precision, recall and alignment error rate (AER) [5].

### 3.1 Basic Word Association Models

Our aim in this section is to compare very simple word association measures reported in the literature and which can be very useful for some applications.

Cherry and Lin [6] and Lambert *et al.* [7] use $\chi^2$ scores [8]. However, Dunning [9] showed that the log-likelihood ratio (LLR) was a better method of accounting for rare events occurring in large samples. $\chi^2$ score indeed overestimates their significance. For example, the association between two singletons cooccurring in the same sentence pair gets the best possible $\chi^2$ score, and this association is 4 orders of magnitude less than the best score according to the LLR statistics. The LLR score was used by Melamed [3] for automatically constructing translation lexicons and by Moore [1] as a word association feature. We compared these association measures to IBM model 1 probabilities [10].

Table 1 shows the alignment results for a basic system composed of the following features: word association, link bonus, unlinked word penalty and two distortion features (counting the number and amplitude of crossing links). The value of the word association feature was calculated as the sum of the word association costs of the links present in the alignment. This cost was simply obtained by taking (minus) the logarithm of respectively the $\chi^2$ score, IBM model 1 probabilities or the LLR score normalised to 1. For IBM model 1 probabilities, we had two features, one for each direction (source-target and target-source).

The substitution of the $\chi^2$ score by the more accurate LLR yielded a 11 points drop in precision.[7] IBM model 1 probabilities are better than association scores and yield a 3.5 points improvement over $\chi^2$ word association scores. Of course, state-of-the-art models like IBM model 4 are expected to perform better.

In lines 1 to 3 of Table 1, the unlinked penalty feature is uniform. In the "IBM1+UM" system, this feature was substituted by a penalty proportional to model 1 NULL link probability, yielding a gain of 2 points in precision and 1 point in recall.

[7] This result may be surprising at first sight. In fact, it makes sense. To take the same example as Moore [11], in our corpus, singletons appearing in each side of the same sentence pair constitute a very significant event. The IBM model 1 probability in this case is actually equal to 1, and the $\chi^2$ score is also the best possible. Although no word can have a higher LLR score with a singleton than another singleton, the LLR score between more frequent words can be much higher. This makes a difference because the alignment hypotheses are expanded with the most probable links first. Thus compared to $\chi^2$, the LLR score gives a relatively higher importance to links involving frequent words, which may be stop words, and a relatively lower importance to links involving less frequent words, which often are content words. Both effects produce noisier alignments.

**Table 1.** Recall (Rs), Precision (Pp) and AER for various types of association scores (for stems) and search strategies. The values shown are the average and standard error (in parentheses) of three feature weights optimisations (from different starting points).

| Line | | Rs | Pp | AER |
|---|---|---|---|---|
| *Score used as association feature (baseline search, one iteration)* | | | | |
| 1 | $\chi^2$ | 62.4 (0.8) | 86.7 (1.5) | 27.1 (0.1) |
| 2 | LLR | 59.4 (0.1) | 75.7 (0.5) | 33.2 (0.3) |
| 3 | IBM1 | 65.9 (0.7) | 90.3 (1.4) | 23.5 (0.3) |
| 4 | IBM1+UM | 67.1 (0.3) | 92.5 (0.4) | 21.9 (0.3) |
| *Source-word-score (SWS) and source-word-position (SWP) searches* | | | | |
| 5 | IBM1+UM, SWS | 67.1 (0.2) | 93.5 (0.5) | 21.6 (0.0) |
| 6 | IBM1+UM, SWP | 66.3 (0.5) | 91.5 (0.4) | 22.8 (0.1) |
| 7 | IBM1+UM, SWP 2 it. | 66.7 (0.5) | 93.2 (0.6) | 21.9 (0.1) |
| 8 | IBM1+UM, SWP 3 it. | 67.3 (0.4) | 93.2 (0.4) | 21.5 (0.1) |

### 3.2 Search

The three beam-search strategies described in Sect. 2 were implemented with dynamic programming and are compared in Table 1 (lines 4, 5 and 6). In the "source-word-position" (SWP) strategy, since alignment hypotheses are expanded at consecutive words, it makes sense to recombine the alignment hypotheses with equal recent history. Although hypothesis recombination helps, this strategy gives the worst results because the first links introduced are not the best ones. The best strategy is "source-word-score" (SWS), in which links to the same words are compared fairly, but keeping the idea of introducing the best links first. This strategy allows to gain 1 point in precision over the baseline, without loss in recall.

In lines 1 to 6, only one iteration of the alignment algorithm was run. Lines 7 and 8 show the effect of running two and three iterations for the SWP search. The initial alignment is the best alignment obtained in the previous iteration. After three iterations, the SWP search achieves comparable performance as SWS after one iteration. SWS and baseline search AER results are actually only improved by 0.2 after the second iteration, and not improved by a third iteration.

## 4 Conclusions

Our results suggest that the log-likelihood ratio is not an adequate word association measure to be used in a discriminative word alignment system. We also observed that even the simplest IBM model probabilities allow a significant improvement of alignment quality with respect to word association measures. Finally, we compared three beam-search strategies. We showed that starting from the empty alignment is not the best choice, and that it is more flexible and accurate to let links to the same word compete together, than to introduce them sequentially in the alignment hypotheses.

## References

1. Moore, R.C.: A discriminative framework for bilingual word alignment. In: Proc. of Human Language Technology Conference. (2005) 81–88
2. Liu, Y., Liu, Q., Lin, S.: Log-linear models for word alignment. In: Proc. of the 43rd Annual Meeting of the Assoc. for Computational Linguistics. (2005) 459–466
3. Melamed, I.D.: Models of translational equivalence among words. Computational Linguistics **26**(2) (2000) 221–249
4. Lambert, P., de Gispert, A., Banchs, R.E., Mariño, J.B.: Guidelines for word alignment evaluation and manual alignment. Language Resources and Evaluation **39**(4) (2005) 267–285
5. Och, F.J., Ney, H.: A systematic comparison of various statistical alignment models. Computational Linguistics **29**(1) (March 2003) 19–51
6. Cherry, C., Lin, D.: A probability model to improve word alignment. In: Proc. of 41th Annual Meeting of the Assoc. for Computational Linguistics. (2003) 88–95
7. Lambert, P., Banchs, R.E., Crego, J.M.: Discriminative alignment training without annotated data for machine translation. In: Proc. of the Human Language Technology Conference of the NAACL. (2007) 85–88
8. Gale, W.A., Church, K.W.: Identifying word correspondences in parallel texts. In: DARPA Speech and Natural Language Workshop. (1991)
9. Dunning, T.: Accurate methods for the statistics of surprise and coincidence. Computational Linguistics **19**(1) (1993) 61–74
10. Brown, P.F., Della Pietra, S.A., Della Pietra, V.J., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics **19**(2) (1993) 263–311
11. Moore, R.C.: On log-likelihood-ratios and the significance of rare events. In: Proc. of Conf. on Empirical Methods in Natural Language Processing. (2004) 333–340