Word Alignment Baselines

John C. Henderson
The MITRE Corporation
202 Burlington Road
Bedford, Massachusetts, USA
jhndrsn@mitre.org

Abstract

Simple baselines provide insights into the value of scoring functions and give starting points for measuring the performance improvements of technological advances. This paper presents baseline unsupervised techniques for performing word alignment based on geometric and word edit distances as well as supervised fusion of the results of these techniques using the nearest neighbor rule.

1 Introduction

Simple baselines provide insights into the value of scoring functions and give starting points for measuring the performance improvements of technological advances. This paper presents baseline unsupervised techniques for performing word alignment based on geometric and word edit distances as well as supervised fusion of the results of these techniques using the nearest neighbor rule.

2 Alignment as binary classification

One model for the task of aligning words in a left-hand-side (LHS) segment with those in a right-hand-side (RHS) segment is to consider each pair of tokens as a potential alignment and build a binary classifier to discriminate between correctly and incorrectly aligned pairs. Any of n source language words to align with any of m target language words, resulting in 2^{nm} possible alignment configurations. This approach allows well-understood binary classification tools to address the problem. However, the assumption made in this approach is that the alignments are independent and identically distributed (IID). This is false, but the same assumption is made by the alignment evaluation metrics. This approach also introduces difficulty in incorporating knowledge of adjacency of aligned pairs, and HMM approaches to word alignment show that this knowledge is important (Och and Ney, 2000).

All of the techniques presented in this work approach the problem as a binary classification task.

2.1 Random baseline

A randomized baseline was created which flips a coin to mark alignments. The bias of the coin is chosen to maximize the F-measure on the trial dataset, and the resulting performance gives insight into the inherent difficulty of the task. If the categorization task was balanced, with exactly half of the paired tokens being marked as aligned, then the precision, recall, and F-measure of the coin with the best bias would have all been 50%. The preponderance of non-aligned tokens shifted the F-measure away from 50%, to the 5-10% range, suggesting that only about 10% of the pairs were aligned. An aligner performing worse than this baseline would perform better by inverting its predictions.

3 Unsupervised methods

There are a number of alignment techniques that can be used to align texts when one lacks the benefit of a large aligned corpus. These unsupervised techniques take advantage of general knowledge of the language pair to be aligned. Their relative simplicity and speed allow them to be used in places where timeliness is of utmost importance, as well as to be quickly tuned on a small dataset.

3.1 Final punctuation

Many LHS segments end in a punctuation mark that is aligned with the final punctuation of the corresponding RHS. A high precision aligner that marks only that alignment is useful for debugging the larger alignment system.

3.2 Length ratios

Short words such as stop words tend to align with short words and long words such as names tend to align with long words. This weak hypothesis is worth pursuit because a similar hypothesis was useful for aligning sen-

	Romanian-English				English-French			
Method	P%	R%	F%	AER%	P%	R%	F%	AER %
random	2.62	2.74	2.68	97.32	11.46	10.99	11.22	88.72
fpunct	100.00	2.92	5.67	94.33	100.00	2.07	4.06	80.27
len (eq. 1)	8.73	29.85	13.51	86.49	18.45	29.32	22.65	78.10
exact	53.55	14.24	22.49	77.51	82.56	3.98	7.59	67.45
wdiag (eq. 4)	23.50	57.89	33.45	66.55	38.56	38.85	38.70	58.27
wedit (eq. 2)	50.49	26.59	34.83	65.17	56.54	7.51	13.26	58.43
lcedit	50.32	26.93	35.08	64.92	56.20	7.62	13.43	58.10
cbox (eq. 7)	30.56	49.74	37.86	62.14	44.53	33.74	38.39	53.14
cdiag (eq. 6)	31.52	49.57	38.53	61.47	45.06	30.66	36.49	53.22
freqratio (eq. 8)	10.53	26.07	15.00	85.00	27.77	10.26	14.98	69.91
$P(L\|R)$ (eq. 9)	9.45	36.54	15.02	84.98	15.72	21.86	18.29	81.41
$P(R\|L)$ (eq. 10)	8.80	16.98	11.59	88.41	13.65	10.26	11.71	81.54
bos (eq. 11)	20.42	20.07	20.24	79.76	35.32	10.65	16.37	59.82
bnnrule	84.88	25.04	38.68	61.32	86.55	8.30	15.14	45.38
nnrule	65.89	63.29	64.57	35.43	35.89	35.43	35.66	58.50

Table 1: Trial set results.

tences (Gale and Church, 1991; Brown et al., 1991). The observation can be codified as a distance between the word at position i on the LHS and the word at position j on the RHS

$$D_{len}(i,j) = 1 - \frac{4 * L(l_i) * L(r_j)}{(L(l_i) + L(r_j))^2} \quad (1)$$

where $L(l_i)$ is the length of the token at position i on the LHS. Note that D_{len} is similar to a normalized harmonic mean, ranging from 0 to 1.0, with the minimum achieved when the lengths are the same. A threshold on D_{len} is used to turn this distance metric into a classification rule.

3.3 Edit distances

The language pairs in the experiments were drawn from Western languages, filled with cognates and names. An obvious way to start finding cognates in languages that share character sets is by comparing the edit distance between words.

Three word edit distances were investigated, and thresholds tuned to turn them into classification rules. D_{exact} indicates exact match with a zero distance and a mismatch with value of 1.0. D_{wedit} is the minimum number of character edits (insertions, deletions, substitutions) required to transform one word into another, normalized by the lengths. It can be interpreted as an edit distance rate, edits per character:

$$D_{wedit}(i,j) = \frac{edits(l_i, r_j)}{L(l_i) + L(r_j)} \quad (2)$$

D_{lcedit} is the same as D_{wedit}, except both arguments are lower-cased prior to the edit distance calculation.

3.4 Dotplot geometry

Geometric approaches to bilingual alignment have been used with great success in both finding anchor points and aligning sentences (Fung and McKeown, 1994; Melamed, 1996). Three distance metrics were created to incorporate the knowledge that all of the aligned pairs use roughly the same word order. In every case, the distance of the pair of words from a diagonal in the dotplot was used.

In the metrics below, the L1 norm distance from a point (i,j) to a line from $(0,0)$ to (I,J) is

$$d_{L_1}(i,I,j,J) = \left| \frac{i}{I} - \frac{j}{J} \right| \quad (3)$$

The first metric, D_{wdiag}, is a normalized distance of the (i,j) pair of tokens to the diagonal on the word dotplot

$$D_{wdiag}(i,j) = d_{L_1}(i, L_w(l), j, L_w(r)) \quad (4)$$

where $L_w(l)$ is the length of the LHS in words.

The next two distances are character based, comparing the box containing aligned characters from the words at position (i,j) with the diagonal line on the character dotplot. Let $L_c(l_i)$ be the number of characters preceding the ith word in the LHS.

Let the left edge of the box be $b_l = L_c(l_i)$, the right edge of the box be $b_r = L_c(l_{i+1})$, the bottom edge of the box be $b_b = L_c(r_j)$, and the top edge of the box be $b_t = L_c(r_{j+1})$. The center of the box formed by the words at (i,j) is

$$(i_c, j_c) = \left(\frac{b_l + b_r}{2}, \frac{b_b + b_t}{2} \right) \quad (5)$$

	Romanian-English				English-French			
Method	P%	R%	F%	AER%	P%	R%	F%	AER %
random	3.44	3.99	3.69	96.31	12.26	12.19	12.22	87.74
fpunct	93.95	3.76	7.23	92.77	99.55	2.55	4.98	80.33
len (eq. 1)	8.90	32.49	13.97	86.03	18.45	29.50	22.70	76.92
exact	44.55	13.84	21.12	78.88	81.92	5.33	10.00	64.19
wdiag (eq. 4)	21.98	60.00	32.17	67.83	39.27	42.62	40.88	56.40
wedit (eq. 2)	41.09	22.35	28.95	71.05	56.45	8.38	14.60	58.86
lcedit	43.02	21.18	28.39	71.61	56.07	8.53	14.81	58.59
cbox (eq. 7)	27.15	48.06	34.70	65.30	41.49	34.40	37.62	55.87
cdiag (eq. 6)	26.93	45.11	33.72	66.28	42.56	31.37	36.12	55.22
freqratio (eq. 8)	10.06	27.35	14.71	85.29	28.47	11.27	16.15	69.12
$P(L\|R)$ (eq. 9)	9.84	29.33	14.74	85.26	15.24	22.81	18.28	80.79
$P(R\|L)$ (eq. 10)	9.64	18.52	12.68	87.32	15.20	12.93	13.97	79.40
bos (eq. 11)	21.77	18.17	19.81	80.19	35.81	12.92	18.99	58.53
bnnrule	79.59	18.84	30.25	69.75	86.99	10.12	18.13	44.19
nnrule	51.67	42.03	46.35	53.65	35.43	35.12	35.27	57.93

Table 2: NON-OFFICIAL test set results (ignoring elements aligned with null).

One character metric is the distance from the center of the character box to the diagonal line of the character dotplot, where $L_c(l)$ is the character length of the entire LHS segment.

$$D_{cdiag}(i,j) = d_{L_1}(i_c, L_c(l), j_c, L_c(r)) \quad (6)$$

The distance of the box to the diagonal line is the second character metric

$$D_{cbox} = \begin{cases} 0 & \text{if diagonal intersects box} \\ \min(\ d_{L_1}(b_l, L_c(l), b_t, L_c(r)), & \text{else} \\ \quad d_{L_1}(b_r, L_c(l), b_b, L_c(r))) & \end{cases} \quad (7)$$

4 Data-driven and supervised methods

The distance metrics and associated classifiers described above were all optimized on the trial data, but they required optimization of at most one parameter, a threshold on the distance. Four metrics were investigated that used the larger dataset to estimate larger models, with parameters for every pair of collocated words in the training dataset.

4.1 Likelihoods

Three likelihood-based distance metrics were investigated, and the first is the relative likelihood of the aligned pairs of words. $c(l_i, LHS)$ is the number of times the word l_i was seen in the LHS of the aligned corpus.

$$D_{freqratio}(i,j) = 1 - \frac{min(c(l_i, LHS), c(r_j, RHS))}{max(c(l_i, LHS), c(r_j, RHS))} \quad (8)$$

The next two are conditional probabilities of seeing one of the words given that the other word from the pair was seen in an aligned sentence. Here RHS_x means the right-hand-side of aligned pair number x in the parallel corpus.

$$P(L|R)(i,j) = P(l_i \in LHS_x | r_j \in RHS_x) \quad (9)$$
$$P(R|L)(i,j) = P(r_j \in RHS_x | l_i \in LHS_x) \quad (10)$$

Note that neither of these is satisfactory as a probabilistic lexicon because they give stop words such as determiners high probability for every conditioning token.

4.2 Bag-of-segments distance

The final data-driven measure that was investigated considers the bag of segments (bos) in which the words appear. The result of the calculation is the Tanimoto distance between the bag of segments that word l_i appears in and the bag of segments that word r_j appears in.

$$D_{bos}(i,j) = \frac{\sum_x |c(l_i, LHS_x) - c(r_j, RHS_x)|}{\sum_x \max(c(l_i, LHS_x), c(r_j, RHS_x))} \quad (11)$$

5 Nearest neighbor rule

The nearest neighbor rule is a well-known classification algorithm that provably converges to the Bayes Error Rate of a classification task as dataset size grows (Duda et al., 2001). The distance metrics described above were used to train a nearest neighbor rule classifier, each metric providing distance in one dimension. To provide comparability of distances in the different dimensions, the distribution of points in each dimension was normalized to have zero mean and unit variance ($\mu = 0, \sigma = 1$). The L2 norm, Euclidean distance, was used to compute distance between points.

Two versions of the nearest neighbor rule were explored. In the first, the binary decisions of the classifiers were used as features, and in the second the distances provided by the classifiers were used as features.

6 Experiments

Two datasets of different language pairs were used to evaluate these measures: Romanian-English and English-French. The measures were optimized on a trial dataset and then evaluated blind on a test set. The Romanian-English trial data was 17 sentences long and the English-French trial dataset was 37 sentences. Additionally, approximately 1.1 million aligned English-French sentences and 48,000 Romanian-English sentences were used for the set of supervised experiments.

Four measures were used to evaluate the classifiers: precision, recall, F-measure, and alignment error rate (AER). Precision and recall are the ratios of matching aligned pairs to the number of predicted pairs and the number of reference pairs respectively. F-measure is the harmonic mean of precision and recall. AER differentiates between "sure" and "possible" aligned pairs in the reference, requiring hypotheses to match those that are "sure" and permitting them to match those that are "possible". (Och and Ney, 2000).

7 Results

Table 1 shows results of the explored methods on the trial data, ordered by degree of supervision and AER on the Romanian-English dataset. The biased coin random aligner is indicated as **random** and the final punctuation aligner is **fpunct**. The classifier based on relative length is **len**. The three edit distance measures are exact match (**exact**), edit distance (**wedit**), and lower-case edit distance (**lcedit**). The geometric measures are word distance to the diagonal (**wdiag**), distance to the character diagonal, (**cdiag**), and distance from the character box made by the word pair to the character diagonal, (**cbox**).

The aligners that take advantage of the training data are below the first horizontal line inside the table. **freqratio** is the classifier based on the relative frequency of the two tokens, $P(L|R)$ aligns words in the LHS with words from the RHS that are often collocated in the training sentences, and the reverse for $P(R|L)$. The bag-of-documents distance classifier is evaluated in **bos**.

The two supervised fusion methods are presented in the final two lines of the file: the binary nearest neighbor rule based on the classification output of the aligners (**bnnrule**), and the nearest neighbor rule based on the distances produced by the aligners (**nnrule**). Both of these results are leave-one-out estimates of performance from the trial set. Note that there is incomplete dominance: the binary representation was superior for English-French and the distance representation was superior for Romanian-English.

Table 2 shows results of the explored methods on the test data. The presented order is the same as the order in Table 1. None of the results varied widely from observations on the trial dataset, suggesting that none of the classifiers were drastically overtrained in the course of optimization on the trial data.

8 Conclusion

Several baseline alignment systems were presented. The individual scores of the different aligners give insight into the relative contributions of the features they exploit. Word length matching appears to be the least important feature, followed by character edit distance (attempting to match cognates), and geometric dotplot distances appear to contribute most strongly to alignment performance.

The supervised probabilistic models perform poorly on their own, probably because of the unconstrained way in which they were trained and applied. When all features are combined in concert into a larger alignment system using the nearest neighbor rule, they perform better than individual aligners, but the question remains of what space should be used for modeling the points (distances versus binary decisions).

References

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. 1991. Aligning sentences in parallel corpora. In *Proceedings of the Annual Meeting of the ACL*, pages 169–176.

R. O. Duda, P. E. Hart, and D. G. Stork. 2001. *Pattern Classification*. John Wiley and Sons Inc.

Pascale Fung and Kathleen McKeown. 1994. Aligning noisy parallel corpora across language groups: Word pair feature matching by dynamic time warping. In *Proceedings of AMTA-94*, pages 81–88, Columbia, Maryland. Association for Machine Translation in the Americas.

William A. Gale and Kenneth W. Church. 1991. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19:75–102.

I. Dan Melamed. 1996. A geometric approach to mapping bitext correspondence. In *Proceedings of the First Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, May.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics.*, pages 440–447, Hong Kong.