Word Alignment by Thresholded Two-Dimensional Normalization

Hamidreza Kobdani, Alexander Fraser, Hinrich Schütze
Institute for Natural Language Processing
University of Stuttgart
Germany
{kobdani,fraser}@ims.uni-stuttgart.de

Abstract

In this paper, we present *2D-Linking*, a new unsupervised method for word alignment that is based on association scores between words in a bitext. 2D-Linking can align m-to-n units. It is very efficient because it requires only two passes over the data and less memory than other methods. We show that 2D-Linking is superior to competitive linking and as good as or better than symmetrized IBM Model 1 in terms of alignment quality and that it supports trading off precision against recall.

1 Introduction

Word alignment, the task of establishing correspondences between words in a bitext (i.e., a sentence-aligned parallel corpus), is an important problem with applications in statistical machine translation, the automatic generation of bilingual dictionaries and cross-language information retrieval. According to Och and Ney (2003), there are two general approaches to computing word alignments: statistical and heuristic methods. In statistical alignment methods (Brown et al., 1993), $Pr\left(T|S\right)$ is written in terms of the conditional probability $Pr\left(T,a|S\right)$ as:

$$Pr\left(T|S\right) = \sum_{a} Pr\left(T,a|S\right) \tag{1}$$

Here, the alignment a describes a mapping from the positions of the words in the source text S to the positions in the target text T.

The heuristic methods are considerably simpler. Generally speaking, they try to align each word according to the associative information of source-target word pairs. This information can be provided using different methods. As our baseline method, we define a simple heuristic model and call it *maximum linking*. For a given target word t_j, maximum linking selects the source word s_i with the highest association score V:

$$s_i = \operatorname*{argmax}_{s'_i \in S} \left\{V(s'_i, t_j)\right\} \tag{2}$$

A refinement of this linking method is competitive linking which removes each linked word pair from the association score matrix (see Section 2).

As another simple heuristic model, we define *θ-linking*. θ-linking selects all links with association scores greater than a threshold θ:

$$\{s_i\} = \left\{s'_i \in S | V(s'_i, t_j) > \theta\right\} \tag{3}$$

The threshold θ helps to keep weak candidates out of the search space.

In this paper we introduce a new method of linking, *Max-θ-Linking*, and a new method for calculating an association score matrix, *two-dimensional normalization* or 2DN. We call the combination of Max-θ-Linking and 2DN *2D-Linking*. The advantages of 2D-Linking are as follows.

- 2D-Linking is very efficient. It can be run on standard hardware for arbitrarily large bitexts. And it can be easily distributed on multiple machines.

- 2D-Linking is more accurate than competitive linking and as accurate or more accurate than IBM Model 1, the current method of choice for initializing training of statistical machine translation models.

- 2D-Linking can align m-to-n units, unlike the IBM models which can only align 1-to-n units.

- 2D-Linking can easily trade off precision vs. recall in word alignment. This is important for many applications of word alignment, in particular for cross-language information retrieval (which in many scenarios requires high precision of word alignments, (Kraaij, 2004)) and machine translation (which usually requires high recall in word alignments, (Fraser and Marcu, 2007b)).

This paper is organized as follows. Related work is discussed in Section 2. Section 3 discusses the three association scores we investigate in this paper: the Dice coefficient, expected mutual information and pointwise mutual information. In Section 4, we describe the 2D-Linking algorithm. Sections 5 and 6 present our evaluation results and conclusions.

2 Related Work

Statistical alignment models depend on a set of unknown parameters that must be learned from training data. IBM Model 1 is a particularly simple instance of the framework presented in Equation 1. This model assumes a uniform prior probability, where all choices of target words generated by a word in the source sentence are equally probable. The translation probability tr $(t_j|s_i)$ of the generated target word depends only on the generating source word. Brown et al. (1993) describe the model as follows:

$$Pr\left(T|S\right) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} \sum_{i=0}^{l} \text{tr}\left(t_j|s_i\right) \qquad (4)$$

Equation 4 is the probability estimate of a target sentence T with length m, given a source sentence S with length l. Due to this equation, IBM Model 1 – like many other word-alignment models – chooses for each target word exactly one source word. For more details see (Brown et al., 1993).

IBM Model 1 is mostly used for initialization of the other IBM models. Structurally, it cannot have m-to-n links. However it is possible to improve the model. Moore (2004) introduced an improvement of IBM Model 1 that tries to control the trade-off between precision and recall by adding additional null words to the source sentence. Like Model 1, Moore's model requires several passes through the data (whereas 2D-Linking only needs two) and cannot model m-to-n links.

There are also some more advanced unsupervised models such as the HMM word alignment model (Vogel et al., 1996), IBM Model 4 (Brown et al., 1993), the Joint (phrase) model (Marcu and Wong, 2002) and Matrix Factorisation (Goutte et al., 2004). But they are much more expensive to calculate. In addition, models like HMM and Model 4 suffer from 1-to-n structure, while the Joint model only allows units consisting of consecutive words. LEAF (Fraser and Marcu, 2007a) directly models m-to-n structure, but is expensive to compute. Alignment by agreement (Liang et al., 2006) is another approach, in which two asymmetric models are trained jointly to maximize a combination of the likelihood of the data. This approach is more expensive than training the two assymetric models independently. It is also biased towards high precision alignment structure as the posterior of both 1-to-n models must be high for a link to be selected.

One of the best known heuristic models is competitive linking (Melamed, 1997). In competitive linking at first the word pair with the highest association score is aligned (this is similar to maximum linking, Eq. 2). Then the corresponding row and column are deleted from the alignment matrix. This process is iterated until either all rows have been deleted or all columns have been deleted. The advantage of this method is its ability to filter out most indirect associations – words that have high association scores but are not translations of each other. The main problem with competitive linking is its inability to produce m-to-n links. Like competitive linking, we will show that 2D-Linking is also able to effectively handle indirect associations. But, importantly, it can find m-to-n links.

2D-Linking is most similar to the approach by Tiedemann (2003). He defines the word alignment clue as a score which indicates an association between source-target words by considering various features derived from co-occurrence statistics. But the search algorithm is more similar to competitive linking though it handles 1-to-n and m-to-1 alignments as well. 2D-Linking could be extended to use Tiedemann's scores in a straightforward manner since it can operate on any association measure.

Och and Ney (2003) and Koehn et al. (2003) defined a heuristic procedure that produces an m-to-n alignment. Start the procedure by generating the predicted 1-to-n alignment in the direction source to target. In this alignment one source word aligns to zero or more target words. Call the resulting alignment A1. Generate the predicted m-to-1 alignment in the direction target to source. In this alignment one target word aligns to zero or more source words. Call the resulting alignment A2. Combine A1 and A2 into an m-to-n alignment using a symmetrization heuristic. We consider the following three symmetrization heuristics in this paper:

(1) The "Union" heuristic takes the union of the links in the A1 and A2 alignments. This results in an alignment having m-to-n discontinuous structure.

(2) The "Intersection" heuristic takes the intersection of the links in the A1 and A2 alignments. This will result in a 1-to-1 alignment structure.

(3) The "Refined" heuristic starts from the "Intersection" 1-to-1 alignment, and adds some of the links present in the "Union" m-to-n discontinuous alignment following the algorithm defined by Och and Ney (2003). This results in an alignment containing 1-to-n and m-to-1 correspondences, but importantly the words in a correspondence must be consecutive, so this is not as general as the "Union" heuristic.

We will show that 2D-Linking has better alignment quality than symmetrized Model 1.

3 Associative Information

In the heuristic models, the way association between words is calculated is of central importance. There are many different methods that can be used, for example Dice coefficient scores (Dice, 1945), point-wise mutual information (PMI) (Manning and Schütze, 1999), log-likelihood-ratio (LLR) (Moore, 2004), or expected mutual information (MI) (Cover and Thomas, 1991). We now discuss the shortcomings of Dice and PMI.

Several approaches to alignment have used the Dice coefficient (e.g., (Zhang et al., 2003)). It is defined as follows (Dice, 1945):

$$Dice\left(s,t\right) = \frac{2C_{ST}(s,t)}{C_S(s)+C_T(t)}$$

where C_S and C_T are the word frequencies in the two languages and C_{ST} is the number of co-occurrences of the two words in sentence pairs of the bitext.

PMI between the source word s and the target word t is calculated as follows:

$$PMI\left(s,t\right) = log_2 \frac{P(s,t)}{P(s)\times P(t)}$$

PMI and Dice are similar because they only consider the number of word occurrences and the number of occurrences of the word pair – but no "non-occurrence" information. Moore (2004) uses the log-likelihood-ratio (LLR):

$$LLR\left(s,t\right) = \sum_{i_s\in\{0,1\}} \sum_{i_t\in\{0,1\}} C\left(s=i_s, t=i_t\right) \times log_2 \frac{P\left(s=i_s|t=i_t\right)}{P\left(s=i_s\right)} \quad (5)$$

The advantage of LLR is that it pays attention to the entire "space" of co-occurrence. That is, Equation 5 considers all cases where the corresponding words occur or do not occur in the respective target and source sentences. This reduces the association scores of word pairs which are not translations of each other (cf. table 1).

Expected mutual information (MI) is a normalized formulation of LLR. MI is calculated as follows: (Cover and Thomas, 1991)

$$MI\left(s,t\right) = \sum_{i_s\in\{0,1\}} \sum_{i_t\in\{0,1\}} P\left(s=i_s, t=i_t\right) \times log_2 \frac{P\left(s=i_s, t=i_t\right)}{P\left(s=i_s\right)\times P\left(t=i_t\right)} \quad (6)$$

We prefer MI over LLR because MI is the standard measure for calculating the mutual dependence of two random variables in information theory.

In table 1, we show Dice, MI and PMI association scores of a number of German words with the English word ***this***, sorted according to MI. The table shows that PMI is correlated with MI. However, its maximum is the ***hauses***, which is not the correct choice. Also, it incorrectly ranks ***haus*** higher than the correct translations ***dieser***, ***diese***, ***dies*** and ***diesen***. Similarly, Dice has ***die*** as maximum and ranks ***ich*** higher than the correct translations ***diesem***, ***dieser***, ***dieses***, ***diese***, ***dies*** and ***diesen***. We attribute the better performance of MI in this case to the fact that it takes into account non-occurrence counts whereas Dice and PMI do not.

German word	trans- lation	MI $\times 10^6$	PMI PMI	Dice $\times 10^3$	
diesem	this	34.11	1.54	9.71	+
dieser	this	28.05	1.31	10.06	+
dieses	this	27.70	1.52	8.29	+
diese	this	17.53	1.02	9.43	+
dies	this	13.74	1.25	5.94	+
diesen	this	6.60	1.05	4.00	+
ich	I	6.30	0.40	12.68	-
der	the	4.41	-0.19	13.62	?
das	the	4.23	0.33	12.25	+
ist	is	4.08	0.34	11.62	-
parlament	parliament	3.85	0.64	5.33	-
namen	names	3.35	-1.45	0.61	-
die	the	3.14	-0.15	14.18	?
thema	subject	3.07	1.08	1.89	-
hier	here	3.04	0.67	4.05	-
haus	house	2.83	1.45	1.07	-
heute	today	2.11	0.65	3.11	-
frage	question	1.98	0.62	3.15	-
wir	we	1.95	0.23	11.16	-
hauses	house	1.86	1.57	0.62	-

Table 1: Association values of the English word ***this*** and its corresponding German words. Translations are marked as correct (+), incorrect (-) or context-dependent (?).

4 2D-Linking

2D-Linking performs two main steps for alignment. To prevent linking of indirect associations, at the first step it normalizes the association scores by dividing each raw score by the sum of the scores of its row and of its column. This produces two sets of normalized probability distributions, one for the rows and one for the columns of the matrix of association scores. These two normalized sets are averaged to produce the new matrix of association scores. The second step of 2D-Linking links the word pairs according to the Max-θ-Linking method.

We now describe 2D normalization (2DN) in detail. Input to 2DN is a matrix M of association scores. We then compute a row-normalized matrix R and a column-normalized matrix C by dividing each element by the sum of its row and its column, respectively:

$$R_{ij} = \frac{M_{ij}}{\sum_{j'=1}^{m} M_{ij'}} \times 100 \qquad (7)$$

$$C_{ij} = \frac{M_{ij}}{\sum_{i'=1}^{l} M_{i'j}} \times 100 \qquad (8)$$

The two matrices are then averaged to build the decision-matrix (D):

$$D_{ij} = \frac{R_{ij} + C_{ij}}{2} \qquad (9)$$

At the linking step, the algorithm uses D to compute the binary word alignment matrix (A) of a sentence pair according to the Max-θ-Linking method as follows.

- As alignment $A_{ij} = 1$ for the source word s_i, choose the target word t_j with the largest association score in D that is greater than the linking threshold θ:

$$A\left[i, \underset{j}{\operatorname{argmax}} \left\{D_{ij} | D_{ij} > \theta\right\}\right] = 1 \qquad (10)$$

- As alignment $A_{ij} = 1$ for the target word t_j, choose the source word s_i with the largest association score in D that is greater than the linking threshold θ:

$$A\left[\underset{i}{\operatorname{argmax}} \left\{D_{ij} | D_{ij} > \theta\right\}, j\right] = 1 \qquad (11)$$

An alternative linking is θ-Linking, which does not impose the maximum constraint:

- As alignments $A_{ij} = 1$ for the source word s_i, choose all target words t_j with the association score in D greater than the linking threshold θ:

$$A\left[i, j\right] = 1 \text{ iff } D_{ij} > \theta \qquad (12)$$

The basic idea of our new method for calculation of association scores is the normalization of columns and rows of association-score matrix and averaging them to build a matrix D of 2DN association scores. In D, the association scores of the indirect associations are de-emphasized, and the association scores of the true translations are emphasized. The example in table 2 shows how this works. In this example, ***she*** must be linked to ***sie***, and ***has*** to ***hat***. By comparing association scores, ***she*** is linked incorrectly to ***hat*** (i.e., indirect association link). But after 2D normalization, the linking score of ***she*** and ***hat*** is reduced and ***she*** is correctly linked to ***sie***.

	sie	hat
she	21	215
has	2	6916

	sie	hat
she	50%	47%
has	4%	98%

Table 2: Example for indirect associations. Scores are $MI \times 10^6$. Left: raw associations, right: associations after 2D normalization.

Example: To illustrate 2D-Linking, consider the German sentence *Ich beziehe mich auf Punkt 11 des Arbeitsplans* ('I refer to item 11 on the order of business.'). table 3 shows the MI matrix M, and table 4 shows the decision matrix D derived from M and the alignment computed by 2D-Linking. Table 5 shows the same sentence pair aligned with competitive linking. An example that shows the advantage of 2D-Linking is the word ***Arbeitsplan***, which is linked to the words ***order*** and ***business***.

Even though the alignment computed by 2D-Linking is better, it is not error-free. The alignment ***auf*** $\leftrightarrow$ ***on*** is an error. Also, ***of*** should probably be linked to ***Arbeitsplan***. Some errors of this type could be fixed by using the word position or phrase alignment, which we briefly discuss in section 6.

5 Evaluation

For the calculation of association scores, we used the Europarl English-German parallel corpus. It consists of about 18.7 million German words, 17.8 million English words and $650{,}000$ sentence pairs.

The linking threshold $(0 \leq \theta \leq 100)$ can be selected to trade off precision and recall. The higher the θ, the higher the precision and the lower the recall.

For the evaluation of 2D-Linking, we use a manually aligned parallel corpus provided to us by Callison-Burch (2007). This gold standard is an annotation of a part of the Europarl English-German parallel corpus. We used 120 sentence pairs which consisted of 3564 English words, 3320 German words and 3223 gold standard links.

We compare the output of the alignment methods with the gold standard using different trade-offs between precision and recall in the F-measure formula as follows: $F_\alpha = 1/(\frac{\alpha}{precision} + \frac{1-\alpha}{recall})$

Table 6 and 7 show the results of our experimental evaluation. We first compare our systems with competitive linking. The best association statistic for competitive linking is PMI (table 6, line 3). Compared with competitive linking, 2D-Linking (Max-θ-Linking + 2DN) results (table 6, lines 25–24) have better F. The largest gap is when we consider F-Measures with small α (i.e. where F-Measure is biased towards recall). The reason for this is that 2D-Linking can create m-to-n alignments, while competitive linking is restricted to 1-to-1 alignments.

Next we compare competitive linking (table 6, lines 15–24) with symmetrized IBM Model 1 (table 6, lines 4–8). At higher values of α, 2D-linking has the same performance as the "refined" Model 1 symmetrization. However as α decreases the story changes and 2D-Linking has superior performance. At $\alpha = 0.4$, "refined" is the best IBM Model 1 symmetrization but it is outperformed by 2D-Linking. At $\alpha = 0.2$, "union" is the best IBM Model 1 symmetrization, but it is also outperformed by 2D-Linking.

We performed additional experiments to try to understand the contributions of the different components of thresholded 2D normalization. The first set of additional experiments we performed compare θ-Linking (table 6, lines 9–14) and Max-θ-Linking (table 6, lines 15–24). Both linking techniques can create m-to-n alignments. For small θs, θ-Linking has higher recall than Max-θ-Linking but at a cost to precision which is too high. For large θs, θ-Linking and Max-θ-Linking have similar results as the filtering by the "maximum rule" (compare Equations 10 and 11 with Equation 12) no longer has much effect. This shows that the maximum rule is effective for generating both high recall and high precision alignments.

We then examined the effect of the association score (2DN) used on 2D-Linking. We will discuss the results for Max-θ-Linking with $\theta = 20\%$, which has a good trade-off between recall and precision. When we tried PMI, we obtained both lower precision and recall than with MI (table 7, lines 1–2). Dice had slightly higher precision and much lower recall than MI (table 7, line 3) . Max-θ-Linking works best with 2DN(MI).

Finally, we also tried competitive linking with 2DN. This did not help performance (table 7, lines 4–5). We believe that the smoothing effect of 2DN is not important when used with competitive linking, given that the competitive linking algorithm deletes

	Ich	beziehe	mich	auf	Punkt	11	des	Arbeitsplan	.
I	**[211.84]**	0.73	**[48.24]**	0.13	2.01	0.01	4.53	0.00	4.78
refer	0.91	**[1.81]**	0.88	2.38	0.04	0.12	0.00	0.13	0.03
to	5.06	0.06	**[1.51]**	**[3.46]**	0.20	0.13	4.00	0.06	0.04
item	0.41	0.09	0.25	0.18	**[3.03]**	0.02	5.84	0.26	0.76
11	0.00	0.17	0.00	0.01	0.14	**[20.92]**	0.02	0.15	0.04
on	0.05	0.16	0.29	**[10.91]**	0.59	0.12	14.92	0.04	2.46
the	1.14	0.00	0.83	0.52	0.09	0.00	**[38.72]**	0.01	4.88
order	0.06	0.01	0.00	0.00	0.02	0.00	0.02	**[0.16]**	0.09
of	4.86	0.00	1.11	0.41	0.18	0.02	**[34.15]**	0.11	2.49
business	0.02	0.11	0.05	0.15	0.00	0.04	0.01	**[0.70]**	0.00
.	8.08	0.01	0.69	0.00	0.62	0.01	3.81	0.01	**[92.59]**

Table 3: The association-score matrix M for the translation example. Scores are expected $MI \times 10^6$. Scores that generate links in table 4 (after normalization) or in table 5 (raw scores) are in bold. Word-by-word translation: **Ich** → I; **beziehe** → refer, relate; **mich** → me, myself; **auf** → on, in, at, by; **Punkt** → point, dot, spot; **des** → of; **Arbeitsplans** → work schedule.

	Ich	beziehe	mich	auf	Punkt	11	des	Arbeitsplan	.
I	**[84.5]**	11.7	**[53.6]**	0.4	14.9	0.0	3.0	0.0	3.1
refer	7.4	**[43.1]**	7.8	25.4	0.7	1.2	0.0	5.0	0.2
to	18.5	1.2	6.6	**[21.5]**	2.1	0.7	15.6	2.1	0.1
item	2.0	1.8	1.4	1.3	**[35.8]**	0.1	29.7	9.3	3.9
11	0.0	3.0	0.0	0.1	1.4	**[97.7]**	0.1	4.8	0.1
on	0.1	2.8	0.8	**[48.5]**	5.2	0.5	32.3	1.3	5.3
the	1.5	0.0	1.7	2.0	0.8	0.0	**[60.2]**	0.4	7.5
order	7.7	1.7	0.5	0.0	3.3	0.0	2.6	**[26.7]**	12.7
of	6.7	0.0	2.3	1.6	1.5	0.1	**[55.5]**	3.5	4.0
business	0.9	7.1	2.3	7.3	0.2	1.9	0.4	**[53.5]**	0.1
.	5.6	0.2	1.0	0.0	4.8	0.0	3.6	0.3	**[86.6]**

Table 4: The decision matrix D for the translation example. The table shows the association scores normalized by 2D-Linking. Scores that are selected as maxima in Equations 10 and 11 and give rise to alignments are in bold.

	Ich	beziehe	mich	auf	Punkt	11	des	Arbeitsplan	.
I	**[211.84]**	0.73	48.24	0.13	2.01	0.01	4.53	0.00	4.78
refer	0.91	**[1.81]**	0.88	2.38	0.04	0.12	0.00	0.13	0.03
to	5.06	0.06	**[1.51]**	3.46	0.20	0.13	4.00	0.06	0.04
item	0.41	0.09	0.25	0.18	**[3.03]**	0.02	5.84	0.26	0.76
11	0.00	0.17	0.00	0.01	0.14	**[20.92]**	0.02	0.15	0.04
on	0.05	0.16	0.29	**[10.91]**	0.59	0.12	14.92	0.04	2.46
the	1.14	0.00	0.83	0.52	0.09	0.00	**[38.72]**	0.01	4.88
order	0.06	0.01	0.00	0.00	0.02	0.00	0.02	0.16	0.09
of	4.86	0.00	1.11	0.41	0.18	0.02	34.15	0.11	2.49
business	0.02	0.11	0.05	0.15	0.00	0.04	0.01	**[0.70]**	0.00
.	8.08	0.01	0.69	0.00	0.62	0.01	3.81	0.01	**[92.59]**

Table 5: The alignment computed by competitive linking for the translation example. Bold scores indicate an alignment of the two corresponding words.

	Alignment Method	F-Measure					
		$\alpha = 0.0$	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1.0$
1	Competitive Linking, DICE	0.30	0.33	0.38	0.43	0.51	0.61
2	Competitive Linking, MI	0.33	0.37	0.41	0.47	0.54	0.64
3	Competitive Linking, PMI	0.40	0.43	0.47	0.52	0.58	0.66
4	IBM Model 1 (Source English)	0.46	0.48	0.50	0.53	0.56	0.59
5	IBM Model 1 (Source German)	0.51	0.52	0.54	0.56	0.57	0.60
6	IBM Model 1 (Union)	0.72	0.52	0.41	0.34	0.29	0.25
7	IBM Model 1 (Intersection)	0.37	0.42	0.48	0.56	**0.67**	0.84
8	IBM Model 1 (Refined)	0.47	0.51	0.55	**0.61**	**0.67**	0.75
9	θ-Linking ($\theta = 0\%$), 2DN(MI)	**0.80**	0.41	0.28	0.21	0.17	0.14
10	θ-Linking ($\theta = 10\%$), 2DN(MI)	0.65	0.61	0.57	0.54	0.51	0.48
11	θ-Linking ($\theta = 20\%$), 2DN(MI)	0.55	0.56	0.58	0.59	0.60	0.62
12	θ-Linking ($\theta = 30\%$), 2DN(MI)	0.46	0.49	0.54	0.58	0.64	0.71
13	θ-Linking ($\theta = 40\%$), 2DN(MI)	0.42	0.46	0.52	0.58	**0.67**	0.78
14	θ-Linking ($\theta = 50\%$), 2DN(MI)	0.34	0.39	0.45	0.53	0.64	0.83
15	Max-θ-Linking ($\theta = 0\%$), 2DN(MI)	0.65	**0.62**	0.59	0.56	0.53	0.51
16	Max-θ-Linking ($\theta = 10\%$), 2DN(MI)	0.62	0.61	**0.60**	0.59	0.58	0.57
17	Max-θ-Linking ($\theta = 20\%$), 2DN(MI)	0.55	0.57	0.59	**0.61**	0.63	0.65
18	Max-θ-Linking ($\theta = 30\%$), 2DN(MI)	0.47	0.51	0.55	0.59	0.65	0.72
19	Max-θ-Linking ($\theta = 40\%$), 2DN(MI)	0.42	0.46	0.52	0.58	**0.67**	0.78
20	Max-θ-Linking ($\theta = 50\%$), 2DN(MI)	0.34	0.39	0.45	0.53	0.64	0.83
21	Max-θ-Linking ($\theta = 60\%$), 2DN(MI)	0.29	0.33	0.40	0.48	0.62	0.87
22	Max-θ-Linking ($\theta = 70\%$), 2DN(MI)	0.22	0.26	0.31	0.40	0.55	0.89
23	Max-θ-Linking ($\theta = 80\%$), 2DN(MI)	0.16	0.19	0.24	0.32	0.47	0.90
24	Max-θ-Linking ($\theta = 90\%$), 2DN(MI)	0.07	0.09	0.11	0.16	0.27	**0.91**

Table 6: F measures for the different combinations of heuristic methods and IBM Model 1. The largest F in each column is in bold. $\alpha = 0$ is recall, and $\alpha = 1$ is precision.

the row and column of each link selected, which at each step is always the maximum cell value of the matrix M.

6 Conclusion

We have presented 2D-Linking, a new unsupervised method of word alignment. In comparison with competitive linking and IBM Model 1, 2D-Linking gives better results. The linking threshold makes it possible to easily trade off precision against recall. Such a tradeoff is important for many applications. For example, cross-language information retrieval requires high precision.

2D-Linking can be based on any definition of association strength, but we have shown that expected mutual information gives better results than using Dice and PMI, two measures that were previously used to compute word alignments.

In 2D-Linking, we can partition the vocabulary and compute association scores for each partition separately. This makes it possible to distribute the process on different machines – or to run the computations sequentially if memory is a scarce resource.

2D-Linking is easy to implement and is sufficiently flexible and modular to be combined with other linguistic or statistical methods.

The m-to-n alignment produced by 2D-Linking can be used directly in many applications of word alignment. However, for bootstrapping a statistical machine translation model such as the IBM models, a probability distribution is needed, rather than a prediction of the best word alignment. This alignment probability distribution can be calculated by normalization of the decision matrix D. In other words, we normalize the association scores of a target word over all its possible translation source words and vice versa to calculate a probability distribution over all lexical connections. This probability distribution can then be used to bootstrap translation models such as the IBM models or a discriminative word alignment model such as (Moore et al., 2006).

In the future we would like to try our alignment model as a lexical knowledge source for computing sentence alignments. We are also planning to ex-

	Alignment Method	Precision	Recall	F Measure ($\alpha = 0.5$)
1	Max-θ-Linking ($\theta = 20\%$), 2DN(MI)	0.65	0.55	**0.60**
2	Max-θ-Linking ($\theta = 20\%$), 2DN(PMI)	0.59	0.43	0.50
3	Max-θ-Linking ($\theta = 20\%$), 2DN(DICE)	0.69	0.31	0.43
4	Competitive Linking, 2DN(PMI)	0.56	0.33	0.42
5	Competitive Linking, 2DN(MI)	0.64	0.33	0.44

Table 7: Precision, recall and F measures for the different combinations of heuristic methods.The largest F is in bold.

tend 2D-Linking to phrase alignment, and in doing this will take advantage of word position. Finally, we intend to use the alignments produced by our enhanced method for building bilingual dictionaries.

Acknowledgments

We thank the DFG for supporting this research through grant SFB 732. We are grateful to Chris Callison-Burch for making his gold standard available to us.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chris Callison-Burch. 2007. *Paraphrasing and Translation*. Ph.D. thesis, University of Edinburgh.

Thomas M. Cover and Joy A. Thomas. 1991. *Elements of Information Theory*. Wiley.

L. R. Dice. 1945. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26:297–302.

Alexander Fraser and Daniel Marcu. 2007a. Getting the structure right for word alignment: LEAF. In *Conference on Empirical Methods in Natural Language Processing and Conference on Computational Natural Language Learning*, pages 51–60.

Alexander Fraser and Daniel Marcu. 2007b. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303.

Cyril Goutte, Kenji Yamada, and Eric Gaussier. 2004. Aligning words using matrix factorisation. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 502, Morristown, NJ, USA. Association for Computational Linguistics.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference*, pages 127–133.

Wessel Kraaij. 2004. *Variations on language modeling for information retrieval*. Ph.D. thesis, University of Twente.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 104–111, New York City, USA, June. Association for Computational Linguistics.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 133–139.

I. Dan Melamed. 1997. A word-to-word model of translational equivalence. In *35th Annual Conference of the Association for Computational Linguistics*.

Robert C. Moore, Wen-Tau Yih, and Andreas Bode. 2006. Improved discriminative bilingual word alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 513–520, Sydney, Australia.

Robert C. Moore. 2004. Improving IBM word-alignment model 1. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*.

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

Jörg Tiedemann. 2003. Combining clues for word alignment. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *16th International Conference on Computational Linguistics*, pages 836–841.

Ying Zhang, Stephan Vogel, and Alex Waibel. 2003. Integrated phrase segmentation and alignment algorithm for statistical machine translation. In *Proceedings of International Conference on Natural Language Processing and Knowledge Engineering*.