Power Mean Based Algorithm for Combining Multiple Alignment Tables

Sameer Maskey, Steven J. Rennie, Bowen Zhou
IBM T.J. Watson Research Center
{smaskey, sjrennie, zhou}@us.ibm.com

Abstract

Alignment combination methods that try to find the best alignment set by combining alignments in two directions are mostly based on heuristics (Och and Ney, 2003), (Koehn et al., 2003). In this paper, we propose a novel mathematical formulation for combining an arbitrary number of alignment tables using their power mean. The method frames the combination task as an optimization problem, and finds the optimal alignment lying between the intersection and union of multiple alignment tables by optimizing the parameter p: the affinely extended real number defining the order of the power mean function. The combination approach produces better alignment tables in terms of both F-measure and BLEU scores.

1 Introduction

Machine Translation (MT) systems are trained on bi-text parallel corpora. One of the first steps involved in training a MT system is obtaining alignments between words of source and target languages. This is typically done using some form of Expectation Maximization (EM) algorithm (Brown et al., 1993), (Och and Ney, 2003), (Vogel et al., 1996). These unsupervised algorithms provide alignment links between english words e_i and the foreign words f_j for a given $e-f$ sentence pair. The alignment pairs are then used to extract phrases tables (Koehn et al., 2003), hierarchical rules (Chiang, 2005), or tree-to-string mappings (Yamada and Knight, 2001). Thus, the accuracy of these alignment links has a significant impact in overall MT accuracy.

One of the commonly used techniques to improve the alignment accuracy is combining alignment tables obtained for source to target ($e2f$) and target to source ($f2e$) directions (Och and Ney, 2003). This combining technique involves obtaining two sets of alignment tables A_1 and A_2 for the same sentence pair $e-f$ and producing a new set based on union $A_\cup = A_1 \cup A_2$ or intersection $A_\cap = A_1 \cap A_2$ or some optimal combination A_o such that it is subset of $A_1 \cup A_2$ but a superset of $A_1 \cap A_2$. How to find this optimal A_o is a key question. $A_\cup$ has high precision but low recall producing fewer alignments and $A_\cap$ has high recall but low precision.

2 Related Work

Most existing methods for alignment combination (symmetrization) rely on heuristics to identify reliable links (Och and Ney, 2003), (Koehn et al., 2003). The method proposed in (Och and Ney, 2003), for example, interpolates the intersection and union of two asymmetric alignment tables by adding links that are adjacent to intersection links, and connect at least one previously unaligned word. Another example is the method in (Koehn et al., 2003), which adds links to the intersection of two alignment tables that are the diagonal neighbors of existing links, optionally requiring that any added links connect two previously unaligned words.

Other methods try to combine the tables during alignment training. In (Liang et al., 2006), asymmetric models are jointly trained to maximize the similarity of their alignments, by opti-

mizing an EM-like objective function based on agreement heuristics. In (Ayan et al., 2004), the authors present a technique for combining alignments based on various linguistic resources such as parts of speech, dependency parses, or bilingual dictionaries, and use machine learning techniques to do alignment combination. One of the main disadvantages of (Ayan et al., 2004)'s method, however, is that the algorithm is a supervised learning method, and so requires human-annotated data. Recently, (Xiang et al., 2010) proposed a method that can handle multiple alignments with soft links which are defined by confidence scores of alignment links. (Matusov et al., 2004) on the other hand frame symmetrization as finding a set with minimal cost using use a graph based algorithm where costs are associated with local alignment probabilities.

In summary, most existing alignment combination methods try to find an optimal alignment set A_o that lies between $A_\cap$ and $A_\cup$ using heuristics. The main problems with methods based on heuristics are:

1. they may not generalize well across language pairs
2. they typically do not have any parameters to optimize
3. most methods can combine only 2 alignments at a time
4. most approaches are ad-hoc and are not mathematically well defined

In this paper we address these issues by proposing a novel mathematical formulation for combining an arbitrary number of alignment tables. The method frames the combination task as an optimization problem, and finds the optimal alignment lying between the intersection and union of multiple alignment tables by optimizing the parameter p of the power mean function.

3 Alignment combination using the power mean

Given an english-foreign sentence pair (e_1^I, f_1^J) the alignment problem is to determine the presence of absence of alignment links a_{ij} between the words e_i and f_j, where $i \leq I$ and $j \leq J$. In this paper we will use the convention that when $a_{ij} = 1$, words e_i and f_j are linked, otherwise $a_{ij} = 0$. Let us define the alignment tables we obtain for two translation directions as A_1 and A_2, respectively. The union of these two alignment tables $A_\cup$ contain all of the links in A_1 and A_2, and the intersection $A_\cap$ contain only the common links. Definitions 1 and 2 below define $A_\cup$ and $A_\cap$ more formally. Our goal is to find an alignment set A_o such that $|A_\cap| \leq |A_o| \leq |A_\cup|$ that maximizes some objective function. We now describe the power mean (PM) and show how the PM can represent both the union and intersection of alignment tables using the same formula.

The power mean:
The power mean is defined by equation 1 below, where p is a real number in $(-\infty, \infty)$ and a_n is a positive real number.

$$S_p(a_1, a_2, ..., a_n) = (\frac{1}{n}\sum_{k=1}^{n} a_k^p)^{\frac{1}{p}} \quad (1)$$

The power mean, also known as the generalized mean, has several interesting properties that are relevant to our alignment combination problem. In particular, the power mean is equivalent to the geometric mean G when $p \to 0$ as shown in equation 2 below:

$$\begin{aligned} G(a_1, a_2, ..., a_n) &= (\prod_{i=1}^{n} a_i)^{\frac{1}{n}} \\ &= \lim_{p\to 0}(\frac{1}{n}\sum_{k=1}^{n} a_k^p)^{\frac{1}{p}} \quad (2) \end{aligned}$$

The power mean, furthermore, is equivalent to the maximum function M when $p \to \infty$:

$$\begin{aligned} M(a_1, a_2, ..., a_n) &= \max(a_1, a_2, ..., a_n) \\ &= \lim_{p\to \infty}(\frac{1}{n}\sum_{k=1}^{n} a_k^p)^{\frac{1}{p}} \quad (3) \end{aligned}$$

Importantly, the PM S_p is a non-decreasing function of p. This means that S_p is lower bounded by G and upper-bounded by M for $p \in [0, \infty]$:

$$G < S_p < M, \quad 0 < p < \infty. \quad (4)$$

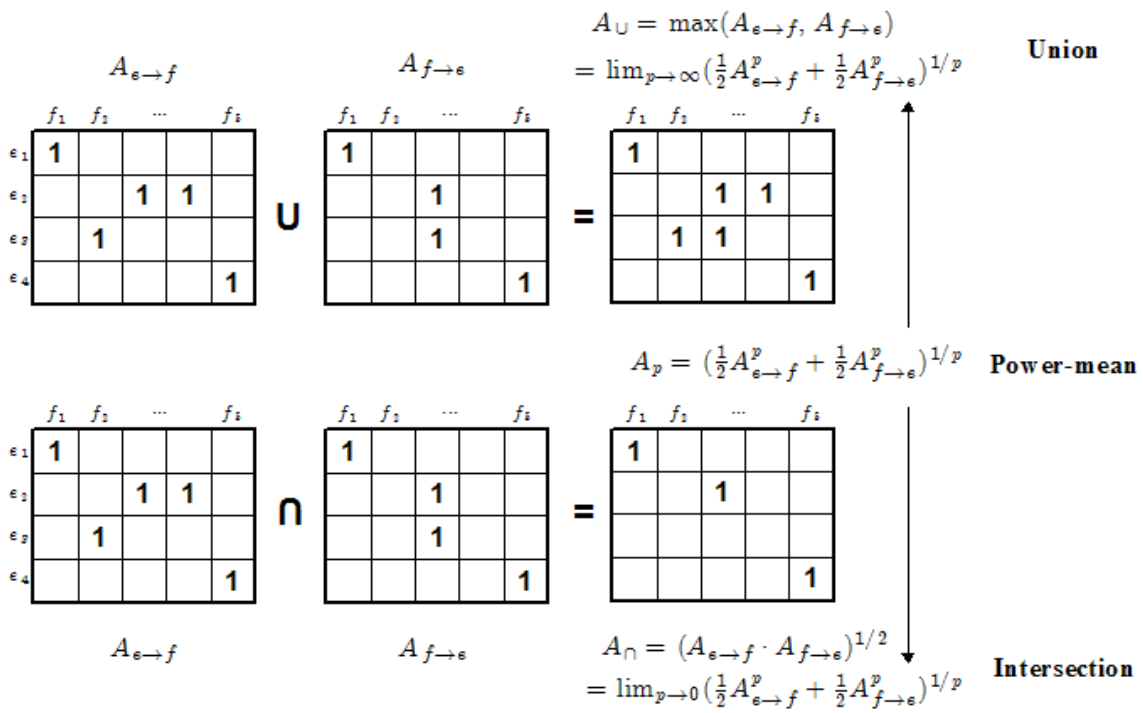

Figure 1: The power-mean is a principled way to interpolate between the extremes of union and intersection when combining multiple alignment tables.

They key insight underpinning our mathematical formulation of alignment combination problem is that geometric mean of multiple alignment tables is equivalent to their logical intersection, while the maximum of multiple alignment tables is equivalent to their logical union. More formally, the union and intersection of two alignment tables is defined as follows.

Definition 1: The union of alignments $A_1, A_2, ..., A_n$ is a set $A_\cup$ that contains a^q_{ij} if $a^q_{ij} = 1$ for any q.

Definition 2: The intersection of alignments $A_1, A_2, ..., A_n$ is a set $A_\cap$ that contains a^q_{ij} if $a^q_{ij} = 1$ for all q.

Figure 1 depicts a simple example of the alignment combination problem for the common case of alignment symmetrization. Two alignments tables, $A_{e\to f}$ and $A_{f\to e}$ (one-to-many alignments), need to be combined. The results of taking the union $A_\cup$ and intersection $A_\cap$ of the tables is shown. $A_\cup$ can be computed by taking the element wise maximum of $A_{e\to f}$ and $A_{f\to e}$, which in turn is equal to the power mean A_p of the elements of these tables in the limit as $p \to \infty$. The intersection of the two tables, $A_\cap$, can similarly be computed by taking the geometric mean of the elements of $A_{e\to f}$ and $A_{f\to e}$, which is equal to the power mean A_p of the elements of these tables in the limit as $p \to 0$. For $p \in (0, \infty)$, equation 4 implies that A_p has elements with values between $A_\cap$ and $A_\cup$. We now provide formal proofs for these results when combining an arbitrary number of alignment tables.

3.1 The intersection of alignment tables $A_1..A_n$ is equivalent to their element wise geometric mean $G(A_1, A_2, ..., A_n)$, as defined in (2).

Proof : Let A_q be any alignment. Let the elements of the A_q be a^q_{ij} such that $a^q_{ij} = 1$ if $\exists$ alignment between the words e_i and f_j and $a^q_{ij} = 0$ otherwise. Let $A_\cap$ be the intersection of the sets A_q where $q \in \{1, 2, .., n\}$. As per our definition of intersection $\cap$ between alignment tables $A_\cap$ contains links where $a^q_{ij} = 1 \; \forall \; q$.

Let A_g be the set that contain the elements of $G(A_1, A_2, ..., A_n)$. Let a^g_{ij} be the geometric mean of the elements a^q_{ij} where $q \in \{1, 2, .., n\}$, as defined in equation 2, that is, $a^g_{ij} = (\prod_{q=1}^n a^g_{ij})^{\frac{1}{n}}$. This product is equal to 1 iff $a^q_{ij} = 1 \; \forall q$ and zero otherwise, since $a^q_{ij} \in 0, 1 \forall \, q$. Hence $A_g = A_\cap$. Q.E.D.

3.2 The union of alignment tables $A_1..A_n$ is equivalent to their element wise maximum $M(A_1, A_2, ..., A_n)$, as defined in (3).

$Proof$: Let a_{ij}^q be an alignment member of A_q and $A_\cup$ be the union of all A_q for $q \in \{1, 2, .., n\}$. As per our definition of the union between alignments $A_\cup$ only contains a_{ij}^q where $a_{ij}^q = 1$ for some q.

Let A_m be the set that contain the elements of $M(A_1, A_2, ..., A_n)$. Let a_{ij}^m be the maximum of the elements a_{ij}^q where $q = 1..n$, as defined in equation (3). The max function is equal to 1 if $a_{ij}^q = 1$ for any q and zero otherwise, since $a_{ij}^q \in \{0, 1\} \forall q$. Hence $A_m = A_\cup$. Q.E.D.

3.3 The element wise power mean $S_p(A_1, A_2, ..., A_n)$ of alignment tables $A_1..A_n$ has entries that are lower-bounded by the intersection of these tables, and upper-bounded by their union for $p \in [0, \infty]$.

$Proof$: We have already shown that the union and intersection of a set of alignment tables are equivalent to the maximum and geometric mean of these tables, respectively. Therefore given that the result in equation 4 is true (we will not prove it here), the relation holds. In this sense, the power mean generalizes the notion of set combination for logical events (and their probabilistic counterparts). Q.E.D.

4 Data

We used the standard English-Pashto data set that was provided to the competing teams of the Darpa Transtac evaluation for all of our experiments. The training data for this task consists of slightly more than 100K parallel sentences. The Transtac task was designed to evaluate speech-to-speech translation systems, so all training sentences are conversational in nature. The sentence length of these utterances varies greatly, ranging from a single word to more than 50 words. 2026 sentences were randomly sampled from this training data to prepare held out development set. The held out Transtac test set consists of 1019 parallel sentences.

5 Experiments and Discussion

We have shown in the previous sections that union and intersection of alignments can be mathematically formulated using the power mean. Since both combination operations can be represented with the same mathematical expression (as a power mean), we can search the combination space "between" the intersection and union of alignment tables by optimizing p w.r.t. any chosen objective function. In the experiments presented here in we define the alignment to be optimal when the function $f(a_{11}, a_{12}, ..., a_{1n}, a_{21}, ..., a_{2n}, a_{n1}, ...a_{nn})$ is maximized, where the function f is standard F-measure. Instead of attempting to optimize the F-measure using heuristics we can now optimize it by finding the appropriate power order p using any suitable numerical optimization algorithm. In our experiments we used the general simplex algorithm of amoeba search (Nelder and Mead, 1965), which attempts to find the optimal set of parameters by evolving a simplex of evaluated points in the direction that the F-measure is increasing.

In order to test our alignment combination formulation empirically we performed experiments on English-Pashto language with data described in Section 4. We first trained two sets of alignments, the e2f and f2e directions, based on GIZA++ (Och and Ney, 2003) algorithm. We then combined these alignments by performing intersection (I) and union (U). We obtained F-measure of 0.5979 for intersection (I), 0.6589 for union (U). For intersection the F-measure is lower presumably because many alignments are not shared by the input alignment tables so the number of links is under-estimated. We then also re-produced the two commonly used combination heuristic methods that are based on growing the alignment diagonally (GDF) (Koehn et al., 2003) and adding links based on refined heuristics (H) (Och and Ney, 2003). We obtained F-measure of 0.6891 for H, and 0.6712 for GDF as shown in Table 1. GDF corresponds to 'diag-and (grow diagonal final)' of (Koehn et al., 2003).

We then used our power mean formulation for combination to maximize the F-measure function

Method	F-measure
I	0.5979
H	0.6891
GDF	0.6712
PM	0.6984
PM_n	0.7276
U	0.6589

Table 1: F-measure Based on Various Alignment Combination Methods

with the aforementioned simplex algorithm for tuning the power parameter p, where F-measure is computed with respect to the hand aligned development data, which contains 150 sentences. This hand aligned development set is different than the development set for training MT models. While doing so we also optimized table weights $W_q \in (0, 1), \sum_q W_q = 1$, which were applied to the alignment tables before combining them using the PM. The W_q allow the algorithm to weight the two directions differently. We found that the F-measure function had many local minima so the simplex algorithm was initialized at several values of p and $\{W_q\}$ to find the globally optimal F-measure.

After obtaining power mean values for the alignment entries, they need to be converted into binary valued alignment links, that is, $S_p(a^1_{ij}, a^2_{ij}, ...a^n_{ij})$ needs to be converted into a binary table. There are many ways to do this conversion such as simple thresholding or keeping best N% of the links. In our experiments we used the following simple selection method, which appears to perform better than thresholding. First we sorted links by PM value and then added the links from the top of the sorted list such that e_i and f_j are linked if e_{i-1} and e_{i+1} are connected to f_j or f_{j-1} and f_{j+1} is linked to e_i or both e_i and f_j are not connected. After tuning power mean parameter and the alignment weights the best parameter gave an F-measure of 0.6984 which is higher than commonly used GDF by 2.272% and H by 0.93% absolute respectively. We observe in Figure 2 that even though PM has higher F-measure compared with GDF it has significantly fewer number of alignment links suggesting that PM has improved precision on the finding the alignment links. The presented PM based alignment combination can be tuned to optimize any chosen objective, so it is not surprising that we can improve upon previous results based on heuristics.

One of the main advantages of the combining alignment tables using the PM is that our statements are valid for any number of input tables, whereas most heuristic approaches can only process two alignment tables at a time. Trying to come up with heuristics for combining more than two tables is error-prone. Heuristics that are designed for N alignment pairs may not be valid for N+k alignments. The presented power mean algorithm in contrast can be used to combine any number of alignments in a single step, which, importantly, makes it possible to jointly optimize all of the parameters of the combination process.

In the second set of experiments the PM approach, which we call PM_n, is applied simultaneously to more than two alignments. We obtained four more sets of alignments from the Berkeley aligner (BA) (Liang et al., 2006), the HMM aligner (HA) (Vogel et al., 1996), the alignment based on partial words (PA), and alignment based on dependency based reordering (DA) (Xu et al., 2009). Alignment I was obtained by using Berkeley aligner as an off-the-shelf alignment tool. We built the HMM aligner based on (Vogel et al., 1996) and use the HMM aligner for producing Alignment II. Producing different sets of alignments using different algorithms could be useful because some alignments that are pruned by one algorithm may be kept by another giving us a bigger pool of possible links to chose from.

We produced Alignment III based on partial words. Pashto is morphologically rich language with many prefixes and suffixes. In lack of a morphological segmenter it has been suggested that keeping only first 'n' characters of a word can effectively reduce the vocabulary size and may produce better alignments. (Chiang et al., 2009) used partial words for alignment training in english and urdu. We trained such alignments using using GIZA++ on parallel data with partial words for Pashto sentences.

The fourth type of alignment we produced, Alignment IV, was motivated by the (Xu et al., 2009). (Xu et al., 2009) showed that transla-

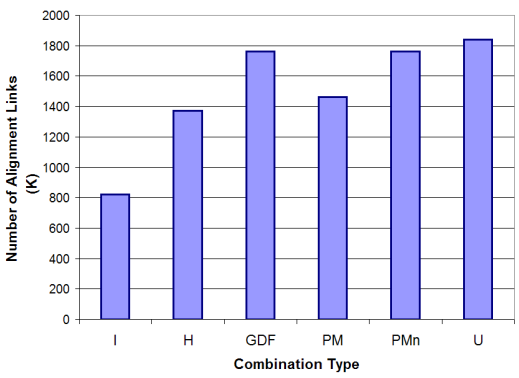

Figure 2: Number of Alignments Links for Different Combination Types

tion between subject-verb-object (English) and subject-object-verb (Pashto) languages can be improved by reordering the source side of the parallel data. They obtained dependency tree of the source side and used high level human generated rules to reorder source side using precedence-based movement of dependency subtrees. The rules were particularly useful in reordering of verbs that moved to the end of the sentence. Making the ordering of source and target side more similar may produce better alignments for language pairs which differ in verb ordering, as many alignment algorithms penalize or fail to consider alignments that link words that differ greatly in sentence position. A Pashto language expert was hired to produce similar precedence-based rules for the English-Pashto language pair. Using the rules and algorithm described in (Xu et al., 2009) we reordered all of the source side and used GIZA++ to align the sentences.

The four additional alignment sets just described, including our baseline alignment, Alignment V, were combined using the presented PM$_n$ combination algorithm, where n signifies the number of tables being combined. We obtained an F-measure of 0.7276 which is 12.97% better than intersection and 6.87% better than union. Furthermore PM$_n$, which in these experiments utilizes 5 alignments, is better than PM by 2.92% absolute. This is an encouraging result because this not only shows that we are finding better alignments than intersection and union, but also that combining more than two alignments is useful. We note that PM$_n$ performed 3.85% better than H (Och and Ney, 2003), and 5.64% better than GDF heuristics.

In the above experiments the parameters of the power mean combination method were tuned on development data to optimize alignment F-measure, and the performance of several alignment combination techniques were compared in terms of Fmeasure. It is not known however if the alignment F-measures correlates well with BLEU scores as explained in (Fraser and Marcu, 2007).

While there is no mathematical problem with optimizing the parameters of the presented PM-based combination algorithm w.r.t. BLEU scores, computationally it is not practical to do so because each iteration would require a complete training phase. To further evaluate the quality of the alignments methods being compared in this paper, we built several MT models based on them and compared the resulting BLEU scores.

E2F	Dev	Test
I	0.1064	0.0941
H	0.1028	0.0894
GDF	0.1256	0.1091
PM	0.1214	0.1094
PM$_n$	0.1378	0.1209
U	0.1062	0.0897

Table 2: E2F BLEU: PM Alignment Combination Based MT Model Comparision

We built a standard phrase-based translation system (Koehn et al., 2003) which utilizes a stack-based decoder based on an A^* search. Based on the combined alignments we extracted phrase tables of maximum length of 6 on English and 8 on Pashto respectively. We then trained the lexicalized reordering model that produced distortion costs based on the number of words that are skipped on the target side, in a manner similar to (Al-Onaizan and Papineni, 2006). Our training sentences are a compilation of sentences from various domains collected by Darpa, and hence we were able to build interpolated language model which weights the domains differently. We built an interpolated LM for both English and Pashto, but for English we had a significantly more monolingual sentences (1.4 million in total) compared

to slightly more than 100K sentences for Pashto. We tuned our MT model using minimum error rate (Och, 2003) training.

F2E	Dev	Test
I	0.1145	0.1101
H	0.1262	0.1193
GDF	0.1115	0.1204
PM	0.1201	0.1155
PM_n	0.1198	0.1196
U	0.1111	0.1155

Table 3: F2E BLEU : PM Alignment Combination Based MT Model Comparision

We built five different MT models based on Intersection (I), Union (U), (Koehn et al., 2003) Grow Diagonal Final (GDF), (Och and Ney, 2003) H refined heuristics and Power Mean (PM_n) alignment sets where $n = 5$. We obtained BLEU (Papineni et al., 2002) scores for E2F direction as shown in Table 2. As expected MT model based on I alignment has the low BLEU score of 0.1064 on dev set and 0.0941 on the test set on E2F direction. Intersection, though, has higher precision, but throws away many alignments, so the overall number of alignments is too small to produce a good phrase translation table. Similarly the U alignment also has low scores (0.1062 and 0.0897) on dev and test sets, respectively. The best scores for E2F direction for both dev and test set is obtained using the model based on PM_n algorithm. We obtained BLEU scores of 0.1378 on dev set and 0.1209 on the test set which is better than all heuristic based methods. It is better by 1.22 absolute BLEU score on dev set and 1.18 on a test compared to commonly used GDF (Koehn et al., 2003) heuristics. The test set BLEU was computed based on 1 reference. We note that for e2f direction PM that combines only 2 alignments is not worse than any of the heuristic based methods. Also the difference in PM and PM_n performance is large signifying that combining multiple alignment helps the power meand based combination algorithm further.

Although we saw significant gains on E2F direction we did not see similar gains on F2E direction unfortunately. Matching our expectation Intersection (I) produced the worse results with

Type	PT Size (100K)
I	182.17
H	30.73
GDF	27.65
PM	60.87
PM_n	25.67
U	24.54

Table 4: E2F Phrase Table Size

BLEU scores of 0.1145 and 0.1101 on dev and test set respectively as shown in Table 3. Our PM_n algorithm obtained BLEU score of 0.1198 on dev set and 0.1196 on test set which is better by 0.83 absolute in dev set over GDF. On the test set though performance between PM_n and GDF is only slightly different with 0.1196 for PM_n and 0.1204 for GDF. The standard deviation on test set BLEU scores for F2E direction is only 0.0042 which is one third of the standard deviation in E2F direction at 0.013 signifying that the alignment seems to make less difference in F2E direction for our models. One possible explanation for such results is that the Pashto LM for the E2F direction is trained on a small set of sentences available from training corpus while English LM for F2E direction was trained on 1.4 million sentences. Therefore the English LM, which is trained on significantly more data, is probably more robust to translation model errors.

Type	PT Size (100K)
I	139.98
H	56.76
GDF	22.96
PM	47.50
PM_n	21.24
U	20.33

Table 5: F2E Phrase Table Size

We should note that difference in alignments also make a difference in Phrase Table (PT) size. Intersection that has the least number of alignments as shown in Figure 2 tend to produce the largest phrase table because there are less restriction on phrases to be extracted. Union tends to produce the least number of phrases because phrase extraction algorithm has more constraints

to satisfy. We observe that PT produced by intersection is significantly larger than others as seen in Tables 4 and 5. The PT size produced by PM_n as shown in Table 4 is between I and U and is significantly smaller than the other heuristic based methods. It is 7.1% smaller than GDF heuristic based phrase table. Similarly in F2E direction as well (Table 5) we see the similar trend where PM_n PT size is smaller than GDF by 4.2%. The decrease in phrase table size but increase in BLEU scores for most of the dev and test sets show that our PM based combined alignments are helping to produce better MT models.

6 Conclusion and Future Work

We have presented a mathematical formulation for combining alignment tables based on their power mean. The presented framework allows us to find the optimal alignment between intersection and union by finding the best power mean parameter between 0 and ∞, which correspond to intersection and union operations, respectively. We evaluated the proposed method empirically by computing BLEU scores in English-Pashto translation task and also by computing an F-measure with respect to human alignments. We showed that the approach is more effective than intersection, union, the heuristics of (Och and Ney, 2003), and the grow diagonal final (GDF) algorithm of (Koehn et al., 2003). We also showed that our algorithm is not limited to two tables, which makes it possible to jointly optimize the combination of multiple alignment tables to further increase performance.

In future work we would like to address two particular issues. First, in this work we converted power mean values to binary alignment links by simple selection process. We are currently investigating ways to integrate the binary constraint into the PM-based optimization algorithm. Second, we do not have to limit ourselves to alignments tables that are binary. PM based algorithm can combine alignments that are not binary, which makes it easier to integrate other sources of information such as posterior probability of word translation into the alignment combination framework.

7 Acknowledgment

This work is partially supported by the DARPA TRANSTAC program under the contract number of NBCH2030007. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

References

Al-Onaizan, Yaser and Kishore Papineni. 2006. Distortion models for statistical machine translation. In *Proceedings of ACL.*

Ayan, Necip, Bonnie J. Dorr, , and Nizar Habash. 2004. Multi-align: Combining linguistic and statistical techniques to improve alignments for adaptable mt. In *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas.*

Brown, P., V. Della Pietra, S. Della Pietra, and R. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chiang, David, Kevin Knight, and Samad Echihabi. 2009. In *Presentation at NIST MT 2009 Workshop, August.*

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of ACL.*

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

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT/NAACL.*

Liang, Percy, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of ACL.*

Matusov, Evgeny, Richard Zens, and Hermann Ney. 2004. Symmetric word alignments for statistical machine translation. In *Proceedings of COLING*, page 219, Morristown, NJ, USA.

Nelder, JA and R Mead. 1965. A simplex method for function minimization. *The Computer Journal 7: 308-313.*

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

Och, Franz J. 2003. Minimum error rate training in statistical machine. In *Proceedings of ACL.*

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *In Proceedings of ACL*, pages 311–318.

Vogel, Stephan, Hermann Ney, and Christoph Tillmann. 1996. Hmm-based word alignment in statistical translation. In *COLING 96: The 16th Int. Conf. on Computational Linguistics*, pages 836–841.

Xiang, Bing, Yonggang Deng, and Bowen Zhou. 2010. Diversify and combine: Improving word alignment for machine translation on low-resource languages. In *Proceedings of ACL*.

Xu, Peng, Jaeho Kang, Michael Ringgaard, and Franz Och. 2009. Using a dependency parser to improve smt for subject-object-verb languages. In *NAACL*, pages 245–253, Morristown, NJ, USA.

Yamada, Kenji and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of ACL*, pages 523–530, Toulouse, France, July. ACL.