Multi-Align: Combining Linguistic and Statistical Techniques to Improve Alignments for Adaptable MT

Necip Fazil Ayan, Bonnie Dorr, and Nizar Habash

Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
{nfa,bonnie,habash}@umiacs.umd.edu

Abstract. The continuously growing MT market faces the challenge of translating new languages, diverse genres, and different domains using a variety of available linguistic resources. As such, MT system adaptability has become a sought-after necessity. An adaptable statistical or Hybrid MT system relies heavily on the quality of word-level alignments of real-world data. Statistical alignment approaches provide a reasonable initial estimate for word alignment. However, they cannot handle certain types of linguistic phenomena such as long-distance dependencies and structural differences between languages (*translation divergences*). We address this issue in Multi-Align, a new framework for incremental testing of different alignment algorithms and their combinations. Our design allows users to tune their systems to the properties of a particular genre/domain while still benefiting from general linguistic knowledge associated with a language pair. We demonstrate that a combination of statistical and linguistically-informed alignments can resolve translation divergences during the alignment process.

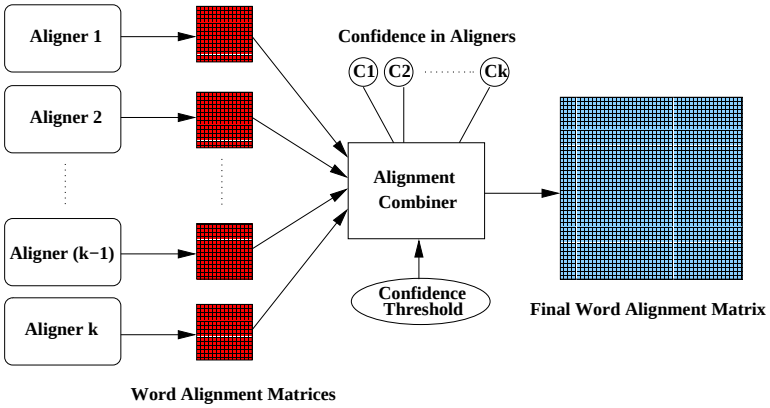
1 Introduction

The continuously growing MT market faces the challenge of translating new languages, diverse genres, and different domains using a variety of available linguistic resources. As such, MT system adaptability has become a sought-after necessity. An adaptable statistical or Hybrid MT system relies heavily on the quality of word-level alignments of real-world data. Statistical alignment approaches provide a reasonable initial estimate for word alignment. However, they simply cannot handle certain types of linguistic phenomena such as long-distance dependencies and structural differences between languages (*translation divergences*) [8].

This paper introduces Multi-Align, a new framework for incremental testing of different alignment algorithms and their combinations. The success of statistical alignment has been demonstrated to a certain extent, but such approaches rely on huge training data to achieve high-quality word alignments. This is not a problem for some language pairs such as English-French. However, data sparseness remains a serious issue for many language pairs, creating a significant bottleneck for statistical systems. Moreover, statistical systems are often incapable of capturing translation divergences, non-consecutive phrasal information, and long-range dependencies. Researchers have addressed these deficiencies by incorporating lexical features into maximum entropy alignment models [20]; however, the range of these lexical features has been limited to simple linguistic phenomena and no results have been reported.

Our alignment combination framework allows users to tune their MT systems to the properties of a particular genre/domain while still benefiting from knowledge of general linguistic phenomena associated with the language pair. Our goal is to induce word-level alignments for MT that are more accurate than those produced by existing statistical systems. This is achieved by combining

Fig. 1. Multi-Align Framework for Combining Different Alignment Algorithms



outputs of different alignment systems. This approach allows us to eliminate the need for rebuilding existing alignment systems to incorporate new linguistic information.

Word-level alignment of bilingual texts is a critical capability for a wide range of NLP applications. In statistical MT, translation models rely directly on word alignments [2]. Construction of bilingual lexicons is a direct application of word alignments, as shown in various studies [16]. In addition, word-level alignments are used for: (1) automatic generation of transfer rules (or mappings) for MT [3]; (2) word sense disambiguation [7]; (3) projection of resources (such as morphological analyzers, part-of-speech taggers, and parsers) from a resource-rich language into other resource-poor languages [12, 26]; and (4) cross-language information retrieval [17]. The quality of word-level alignments plays a crucial role in the success of these applications. For example, in statistical machine translation, it has been shown that improved word alignment—a central goal of the framework described below—directly affects the output quality of statistical MT systems [21].

The next section presents Multi-Align, a framework for producing improved alignments through the combination of different word-alignment models. Section 3, describes a feasibility experiment using two alignment approaches: GIZA++ [19] and DUSTer [8]. Section 4 demonstrates the effect of combining statistical and linguistically-informed knowledge on resolving translation divergences during word alignment. Finally, Section 5 describes future work on extension of Multi-Align to various linguistic resources.

2 Multi-Align

Multi-Align is a general alignment framework where the outputs of different aligners are combined to obtain an improvement over the performance of any single aligner. This framework provides a mechanism for combining linguistically-informed alignment approaches with statistical aligners.

Figure 1 illustrates the Multi-Align design. The output of each word-alignment algorithm is associated with a confidence score indicating its reliability and an *Alignment Combiner* uses this to generate a single word-alignment output. The contribution of each aligner is proportional to its confidence score. The decision to include a particular pair of words in the final alignment is based on a human-specified or machine-learned threshold value.

The basic data structure in Multi-Align is the *word alignment matrix*, Z, similar to that of current statistical-alignment approaches [22]. Z is a $(m+1) \times (n+1)$ matrix, where m is the number of words in the English sentence and n is the number of words in the foreign-language

(FL) sentence.[1] Each entry of the matrix, Z_{st}, corresponds to the alignment information between s^{th} English word and t^{th} FL word. Rather than using binary values for Z_{st}, we use an alignment probability for each Z_{st} of the matrix.

Formally, we assume that k different aligners are used to generate the word alignments between two sentences. Let A^i be the i^{th} alignment system, W^i be the word alignment matrix generated by this system, and C^i be the confidence value of this system. Each entry of W^i, say W^i_{st}, is given a probability of p^i_{st}, showing the probability that the corresponding words (the s^{th} English word and t^{th} FL word) are aligned together. The contribution of the aligner A^i to a specific entry $[s,t]$ of the final word alignment matrix is $C^i \times p^i_{st}$. Thus,

$$Z_{st} = \frac{\sum_{i=1}^{k} C^i \times p^i_{st}}{\sum_{i=1}^{k} \sum_{j=0}^{n} C^i \times p^i_{sj}}$$

Note that the denominator is used to normalize the probabilities over all the probabilities corresponding to the s^{th} English word. For a given confidence threshold ϕ, the s^{th} English word and the t^{th} FL word are aligned together if $Z_{st} > \phi$.

The model includes the parameters, W^i, p^i_{st}, C^i and ϕ, to decide whether two words are aligned to each other or not in this framework. Here are some guidelines for setting these parameters:

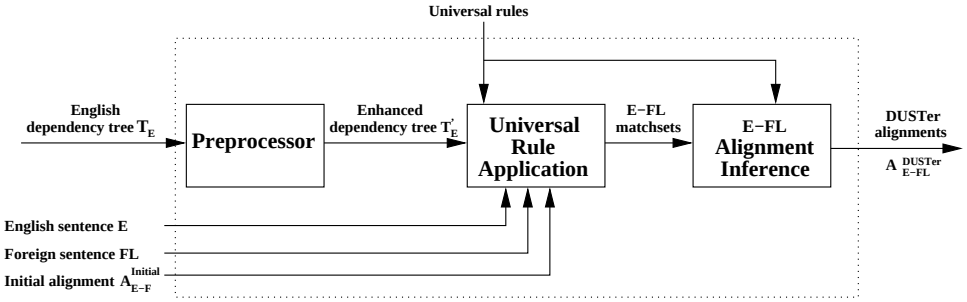
- **Generating W^i, p^i_{st}**: If the aligner produces a set of probabilities for every pair of words between the sentences, it is sufficient to set $W^i_{st} = p^i_{st}$ for all pairs of $[s,t]$. If the aligner provides a list of alignment links with no associated probabilities, $W^i_{st} = 1$ if (s,t) is an alignment link and $W^i_{st} = 0$ otherwise.
- **Setting C^i**: This is a value associated with the confidence of a particular aligner. If all aligners are taken to be equivalent (i.e., there is no priority of one aligner over another), all C^i values are set to the same value, e.g., 1. If one of the aligners is taken to be superior to the others, the confidence value C^i for this particular aligner is set to a higher value than that of other aligners. Alignment confidence values may be human specified or set automatically using machine learning techniques.
- **Setting ϕ**: This is a value associated with the threshold on alignment confidence that is dependent on two factors: (1) whether one-to-many alignments are allowed or not; and (2) maintaining a balance between confidence values of the aligners. If the alignments are forced to be one-to-one, then only the entry with the highest probability in each row and column is deemed *correct*. If all aligners are treated equally, setting $\phi = 0$ is sufficient. Again, this confidence threshold may be human-specified or estimated using machine learning algorithms.

Multi-Align has three advantages with respect to adaptable MT systems: ease of adaptability, robustness, and user control.

- **Ease of Adaptability:** Multi-Align eliminates the need for complex modifications of pre-existing systems to incorporate new linguistic resources. A variety of different statistical and symbolic word alignment systems may be used together. The following are some examples:
  - Statistical alignments [19].
  - Bilingual dictionaries acquired automatically using lexical correspondences [13, 16].
  - Lists of predefined word-pairs, e.g., closed-class words (pronouns or determiners) [1].
  - Set of cognates [24].
  - Syntactic trees on either side [25].
  - Dependency trees on either side [5].

[1] The additional column (Z_{s0} for $0 \leq s \leq m$) and the row (Z_{0t} for $0 \leq t \leq n$) in the matrix is for unaligned English and unaligned FL words, respectively.

Fig. 2. Application of DUSTer's Universal Rules to Infer Missing Alignment Links



- Phrase-based alignments [14].
- Linguistically-motivated alignments [8].

– **Robustness:** Individual alignment systems have inherent deficiencies that result in partial alignments in their output. For example, statistical models have a one-to-one alignment restriction [2, 16]. Multi-Align relies on the strengths of certain systems systems to compensate for the weaknesses of other systems.
– **User Control:** The effect of different linguistic information is difficult to observe and control when linguistic knowledge is injected into statistical maximum entropy models [20]. Multi-Align avoids this problem by helping users to understand which linguistic resources are useful for word alignment. Additionally, the contribution of each aligner may be weighted according to its impact on the target application. For example, if the end application is bilingual-lexicon construction, the alignment of function words may not be as important as the alignment of high-content words. On the other hand, if the end application is statistical machine translation, the alignments should be as complete as possible.

3 Feasibility of the Multi-Alignment Approach: Combining Statistical and Linguistically Informed Alignments

This section describes feasibility experiments we conducted to study the combination of both statistically-induced and linguistically-motivated alignments. Previous work demonstrated that mapping words into word classes [18] is useful for statistical alignment. We take this one step further by using semantic-based word classes and a set of general, linguistically-motivated rules from *DUSTer (Divergence Unraveling for Statistical Translation)* [8] to induce alignment improvements over GIZA++ (a state-of-the-art alignment system).

A central claim of this endeavor is that linguistic information is critical to the overall advancement of non-linguistic approaches to alignment. Our own error analysis of statistically-induced alignments—the output of GIZA++ on 99 English-Spanish sentences—reveals that 80% of statistical alignment errors correspond to missing alignment links. A deeper analysis also shows that 61% of these missed alignments are related to verbs, functional nouns and obliques, which form the main categories of words that are handled by linguistically-motivated components such as the universal rules of DUSTer.

In our feasibility experiment, we combine DUSTer knowledge (i.e., parameterized universal rules) with statistically-induced alignments to produce a complete set of final alignment links. As shown in Figure 2, the input to DUSTer is an English sentence E, a foreign language sentence FL, a dependency tree T_E corresponding to E, and a set of initial word alignments from E to FL,

$A_{E-FL}^{Initial}$. The initial alignments may be produced by any existing automatic alignment system and the dependency tree may be produced by a standard parser.[2] The key idea is to relate one or more linguistically-motivated categories associated with the (English) input words to those of another language (FL); the resulting *match sets* are used to infer additional alignments from an initial set of statistically-induced alignments.

The remainder of this section presents the linguistic knowledge associated with DUSTer, after which an example of rule application and alignment inference is given. Finally, we describe the combination of DUSTer and GIZA++ in Multi-Align.

3.1 Parameters

DUSTer's universal rules require certain types of words to be grouped together into *parameter classes* based on semantic-class knowledge, e.g., classes of verbs including *Aspectual, Change of State, Directional*, etc. [15]. The parameter classes play an important role in identifying and handling language divergences. The current classification includes 16 classes of parameters. Because the parameters are based on semantic knowledge, the English values can be projected to their corresponding values in a new language. With few exceptions, this can be done simply by translating the words of a parameter class to those of another language. For example, the English light verbs *be, do, give, have, make, put, take* are translated to Spanish light verbs *estar, ser, hacer, dar, tomar, poner, tener*, respectively.[3]

3.2 Universal Rule Set

Universal rules relate one or more linguistically-motivated categories in English—specifically, part-of-speech (POS) labels and semantic word classes—to those of a foreign language. For example, the rules in Figure 3 can be used to handle two forms of conflation (Tense-Verb and Light-Verb). These rules correspond to the mappings *'will eat'* → *'eats'* and *'j fears k'* → *'j has fear of k'*, respectively. The first line shows the rule name, the languages to which the rule may be applied, and the order of the nodes in the surface form for each language involved. The ordering numbers (1, 2, 3, ...) correspond to the node identifiers in the rule specified in the subsequent lines. Each rule relates the English dependency tree structure on the left-hand side (LHS) to the foreign-language tree structure on the right-hand side (RHS).[4]

Each node in the rule specification contains a *node type*—a part-of-speech category (POS) or a parameter type—followed by a list of features. The feature list includes a unique *node identifier* (an integer, e.g., *1*), an *alignment index* that relates potentially aligned LHS and RHS nodes (a letter of the alphabet, e.g., *j*), and categorial-variation information (e.g., *CatVar:AJ_N*).[5] For example, the node [*Noun<2,j,Subj>*] in rule 1.B.X specifies the type *Noun*, the identifier 2, the alignment index *j*, and the relation *Subject* with respect to the head node (i.e., the dominating $Psych - Verb$). Because nodes with the same alignment index are viewed as translational equivalents, they are assumed to be potentially aligned in the initial alignments. Note that, in addition to the simple indices, *i*, *j*, etc., we make use of *conflated indices* (*C:i*, *C:j*, etc.) to refer to semantically light

[2] Currently, the initial alignments are produced by GIZA++ based on IBM Model 4 [2], and the dependency trees are generated using Collins' parser [6].

[3] It is not necessary to list all morphological variants of the same word in the parameter classes; different morphological variants are mapped to the root word in a separate module and each variant is treated as a member of the associated parameter class during the execution of the system.

[4] The current set of universal rules supports 4 foreign languages: Spanish, Hindi, Chinese, and Arabic. There are 21 rules for Hindi, 28 rules for Spanish, 44 rules for Arabic and 65 rules for Chinese.

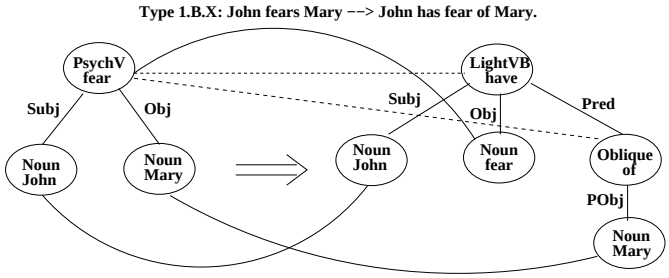
[5] Categorial variations are relations between words like *jealous* (an adjective) and *jealousy* (a noun). These are extracted from a large database called CatVar [11].

Fig. 3. Examples of DUSTer's Universal Rules

```
0.AVar.X [English{2 1} Chinese{1} Spanish{1} Hindi{1} ]
[Verb<1,i> [TenseV<2,Mod,Verb,C:i>]] <--> [Verb<1,i>]

1.B.X [ English{2 1 3} Spanish{2 1 3 4 5} ]
[PsychV<1,i,CatVar:V_N,Verb> [Noun<2,j,Subj>] [Noun<3,k,Obj>]] <-->
 [LightVB<1,Verb,C:i> [Noun<2,j,Subj>] [Noun<3,i,Obj>]
 [Oblique<4,Pred,Prep,C:i> [Noun<5,k,PObj>]]]
```

Fig. 4. Universal Rule Application Example



words that co-occur with high-content words but are generally unaligned in the initial alignments. Nodes marked *C:i* are taken to be related structurally to a (single) high-content node marked i.[6]

Figure 4 illustrates the relation between the LHS and RHS of the second rule in Figure 3. Solid lines between the two sides indicate the alignments that are assumed to be available in the initial alignments (i.e., nodes marked with simple indices i, j, etc.). Dashed lines indicate the alignment links that must be added to produce the final alignment (i.e., nodes marked with *C:i*).[7]

3.3 Application of Universal Rules for Alignment Inference

This section describes how the universal rules and parameters presented above are used to infer alignments from a sentence pair, English dependency tree, and an initial alignment, as in the example of Figure 5.

The dependency tree T_E, produced initially by the Collins parser [6], is augmented during the preprocessing step (the first module in Figure 2) with semantic parameters and CatVar information; the result is the augmented tree T'_E as shown in Figure 5. The enhanced dependency T'_E is next passed to the Universal-Rule Application component, along with the original E and FL sentences and initial alignments $A^{Initial}_{E-FL}$. The initial alignments $A^{Initial}_{E-FL}$ are used as a strict filter on the application of the universal rules. Specifically, we require that all LHS/RHS nodes related by a simple index (i.e., i, j) have a corresponding alignment link in the initial alignments. Moreover, unindexed nodes must **not** have a corresponding alignment link in the initial alignment. Rules violating this requirement are eliminated. The remaining (potentially-applicable) rules are checked

[6] The rationale for distinguishing between these two types of indices is that the semantically-light words (corresponding to nodes marked *C:i*) are generally unaligned in the initial word-alignment process, whereas the co-ocurring high-content word (corresponding to the node marked i) usually has an initial alignment link.

[7] It is important to note that this approach does not require a FL dependency tree as input. To test the applicability of a specific rule, the English dependency tree and surface relative order are matched against the LHS of the rule and the FL surface string is matched against the RHS side of the rule. If both conditions are satisfied, then the rule is applicable.

Fig. 5. Example Sentences, Dependency Tree and Initial Alignment

English Sentence (E): *She will fear her enemies .*					
Spanish Sentence (FL): *Ella tendrá miedo de sus enemigos .*					
Enhanced English Dependency Tree T'_E					
NODE	**WORD**	**POS**	**PARENT**	**REL**	**FEATURES**
1	She	Noun	3	Subj	[FunctionalN:1]
2	will	Verb	3	Mod	[TenseV:1 CatVar:V_N CatVar:V_AJ]
3	fear	Verb	*root*	*	[PsychV:1 CatVar:V_N CatVar:V_AJ CatVar:V_AV]
4	her	Noun	5	Mod	[FunctionalN:1]
5	enemies	Noun	3	Obj	[CatVar:N_AJ]
Initial Alignment $A_{E-FL}^{Initial}$**:** (She, Ella), (fear, miedo), (her,sus), (enemies, enemigos)					

for a match against the POSs and parameter labels associated with the words in the two input sentences.

In addition to checking for a match, the rule-matching process identifies specific sentence positions of matching tokens. That is, given a modified (English/FL) sentence pair and a specific rule, the rule-application module returns the *match sets* corresponding to positions of words that match the RHS and LHS nodes of the rule.[8] In the example given in Figure 5 the match sets resulting from the application of rules 0.AVar.X and 1.B.X are (([2,3],[2])) and (([1,3,5],[1,2,3,4,6])), respectively.

The final step is alignment inference, a straightforward extraction of corrected alignment pairs A_{E-FL}^{DUSTer} using the match sets. Formally, for each match-set element ([. . .,e_i,. . .] [. . .,f_j,. . .]), where e_i and f_j carry the same coindex k (or the same conflated coindex), we add the pair (e_i,f_j) to the partial alignments.

3.4 Combining GIZA++ and DUSTer in MultiAlign

Our feasability experiment combines GIZA++ [19] and DUSTer (described above). GIZA++ is based on IBM Model 4 [2]. For generating GIZA++ alignments, we use the default parameters that are provided with the package. More specifically, the alignments are bootstrapped from Model 1 (five iterations), HMM model (five iterations), Model 3 (two iterations) and Model 4 (four iterations).

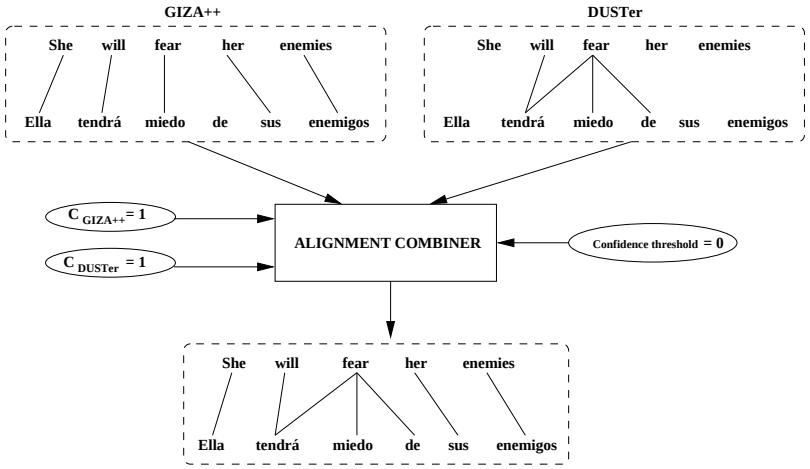
GIZA++ and DUSTer are given as input alignment systems in the Multi-Align framework. Since DUSTer provides only partial alignments that are related to the translation divergences, we are interested in a union of these two systems' outputs to produce the final set of alignments. Therefore, the confidence value for each aligner is set to 1 and the confidence threshold is set to 0. Figure 6 shows the set of alignments generated by GIZA++ and DUSTer and their combination in MultiAlign for the sentence pair in our example.

4 Results

In order to evaluate our combined approach, we conducted an experiment to compare the alignments produced in the Multi-Align framework to those produced by GIZA++ (the industry standard) alone.[9] We measure the closeness of Multi-Align and GIZA++ output to alignments produced manually for a (held-out) test set of 100 Spanish/English sentence pairs using precision, recall and f-measure metrics.

[8] Because the rule may match the input in more than one place, the match sets are stored as lists of lists.
[9] GIZA++ was trained on an English-Spanish training corpus of 45K sentence pairs.

Fig. 6. Set of Alignments Generated by Using GIZA++ and DUSTer in Multi-Align



The unit of comparison is the alignment pair, i.e., the English word and its corresponding (aligned) FL word. Let A be the alignments generated by either the combined approach or GIZA++ and let G be the gold standard alignments. Each element of A and G is a pair (e_i, f_j) where e_i is an English word and f_j is a FL word. Many-to-many alignments are allowed, i.e., either A or G may include more than one pair containing the same English word or FL word. Formally, precision, recall and f-measure metrics are defined as follows:

$$Recall(R) = \frac{|A \cap G|}{|G|} \qquad Precision(P) = \frac{|A \cap G|}{|A|} \qquad F - measure = \frac{2 \times P \times R}{P + R}$$

Table 1 summarizes the evaluation results for both sets. The difference between the Multi-Align and GIZA++ scores are statistically significant at a 95% confidence level using a two-tailed t-test. A one-tailed t-test also shows that Multi-Align is significantly better than GIZA++ using all 3 measures.

Table 1. Spanish/English Alignment Evaluation: DUSTer and GIZA++

SYSTEM	PRECISION	RECALL	F-MEASURE
GIZA++	72.32	72.68	72.43
Multi-Align	73.01	73.36	73.12

5 Conclusion and Future Work

We have introduced a general framework, Multi-Align, to combine outputs of different word alignment systems for an improved word alignment. To illustrate the effectiveness of the framework, we conducted a feasibility experiment that combines linguistically-motivated and statistically-induced alignments to resolve structural differences between languages to add missing alignment links.

While our result appears to be a modest improvement, we have used the strictest possible application of the universal rules to obtain these results. In particular, we have required that all LHS/RHS nodes carrying a simple index have a corresponding alignment link in the initial alignments—and also that unindexed nodes **not** have a corresponding alignment link in the initial alignment. In future work, we will investigate using a bilingual dictionary and/or a POS tagger for

foreign language in order to obtain better match sets on the FL side. More accurate match sets on both sides will allow us to correct some of the added and/or replaced alignments by statistical systems as well as adding alignments missed by those systems.

The most critical task in the framework is the setting of the Multi-Align parameters of the framework appropriately to decrease/increase the effects of a particular alignment system. In practice, the user may not foresee the quality of word alignments produced by a particular alignment system, therefore it may not be practical to set these parameters manually. We will investigate estimation of these parameters, i.e., confidence values and confidence threshold, automatically using machine learning techniques.

The improved alignments are most useful when seen in the context of the larger goals behind our alignment work: (1) to train foreign language parsers using the projected dependency trees and (2) to improve the performance of both statistical and hybrid symbolic/statistical MT systems. Our next step is to conduct an external evaluation of our divergence-unraveling approach based on a quality assessment of the projected dependency trees used for parser training and the output of the trained parser. We will compare the resulting FL dependency trees with human-produced FL treebanks, e.g., Chinese and Arabic treebanks [4, 9]. The goal of improving the performance of hybrid symbolic/statistical MT systems is addressed by embedding the output of our (projected dependency-tree) trained parser in a Generation-Heavy MT (GHMT) framework [10]. We will investigate this path and conduct evaluations using the standard MT metrics, e.g., BLEU [23].

Acknowledgements

This work has been supported, in part, by ONR MURI Contract FCPO.810548265, Cooperative Agreement DAAD190320020, NSF ITR Grant IIS-0326553, and NSF Infrastructure Award EIA0130422. We are indebted to several researchers (Andrew Fister, Ayelet Goldin, Rebecca Hwa, Nitin Madnani, Eric Nichols) who assisted in developing DUSTer Version 1.0 (the E' model), which served as a crucial step toward the development of the tighter, more direct design reported here.

References

1. Lars Ahrenberg, Mikael Andersson, and Magnus Merkel. A Simple Hybrid Aligner for Generating Lexical Correspondences in Parallel Texts. In *ACL/COLING 1998, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (joint with the 17th International Conference on Computational Linguistics)*, Montreal, Canada, August 10–14 1998.
2. Peter F. Brown, Stephan A. Della-Pietra, and Robert L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311, 1993.
3. Jaime Carbonell, Katharina Probst, Erik Peterson, Christian Monson, Alon Lavie, Ralf Brown, and Lori Levin. Automatic Rule Learning for Resource-Limited MT. In *AMTA 2002, Proceedings of the Fifth Conference of the Association for Machine Translation in the Americas*, Tiburon, California, 2002.
4. John Carroll, Ted Briscoe, and Antonio Sanfilippo. Parser Evaluation: a Survey and a New Proposal. In *LREC 1998, Proceedings of the First International Conference on Language Resources and Evaluation*, pages 447–454, Granada, Spain, May 1998.
5. Colin Cherry and Dekang Lin. A Probability Model to Improve Word Alignment. In *ACL 2003, Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 88–95, July 2003.
6. Michael Collins. Three Generative Lexicalized Models for Statistical Parsing. In *ACL 1997, Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, Madrid, Spain, 1997.
7. Mona Diab and Philip Resnik. An Unsupervised Method for Word Sense Tagging Using Parallel Corpora. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, July 2002.

8. Bonnie J. Dorr, Lisa Pearl, Rebecca Hwa, and Nizar Habash. DUSTer: A Method for Unraveling Cross-Language Divergences for Statistical Word–Level Alignment. In *AMTA 2002, Proceedings of the Fifth Conference of the Association for Machine Translation in the Americas*, Tiburon, California, 2002.
9. Joshua Goodman. Parsing Algorithm and Metrics. In *ACL 1996, Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, California, June 24–27 1996.
10. Nizar Habash. Generation Heavy Hybrid Machine Translation. In *INLG 2002, Proceedings of the International Workshop on Natural Language Generation*, New York, NY, 2002.
11. Nizar Habash and Bonnie J. Dorr. A Categorial Variation Database for English. In *NAACL/HLT 2003, Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference*, pages 96–102, Edmonton, Canada, May–June 2003.
12. Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. Evaluating Translational Correspondence Using Annotation Projection. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 392–399, Philadelphia, Pennsylvania, July 2002.
13. Sue J. Ker and Jason S. Chang. A Class-based Approach to Word Alignment. *Computational Linguistics*, 23(2):313–343, 1997.
14. Philip Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *NAACL/HLT 2003, Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference*, Edmonton, Canada, May 27–June 1 2003.
15. Beth Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL, 1993.
16. I. Dan Melamed. Models of Translational Equivalence Among Words. *Computational Linguistics*, 26(2):221–249, 2000.
17. D. W. Oard and B. J. Dorr. A Survey of Multilingual Text Retrieval. Technical report, University of Maryland, Institute for Advanced Computer Studies, April 1996. Technical Report UMIACS-TR-96-19.
18. Franz Joseph Och. An Efficient Method to Determine Bilingual Word Classes. In *EACL 1999, Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, Bergen, Norway, June 1999.
19. Franz Joseph Och. Giza++: Training of Statistical Translation Models. Technical report, RWTH Aachen, University of Technology, 2000. Available at http://www-i6.informatik.rwthaachen.de/õch/software/GIZA++.html.
20. Franz Joseph Och and Hermann Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, Pennsylvania, July 2002.
21. Franz Joseph Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):9–51, March 2003.
22. Franz Joseph Och and Hans Weber. Improving Statistical Natural Language Translation with Categories and Rules. In *ACL/COLING 1998, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (joint with the 17th International Conference on Computational Linguistics)*, pages 985–989, Montreal, Canada, August 10–14 1998.
23. Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, July 2002.
24. Michel Simard, G. Foster, and P. Isabelle. Using Cognates to Align Sentences in Bilingual Corpora. In *TMI 1992, Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 67–81, Montreal, Canada, June 1992.
25. Kenji Yamada and Kevin Knight. A Syntax-Based Statistical Translation Model. In *ACL 2001, Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, 2001.
26. David Yarowsky, Grace Ngai, and Richard Wicentowski. Inducing Multilingual Text Analysis Tools via Robust Projection Across Aligned Corpora. In *HLT 2001, Proceedings of the Human Language Technology Conference*, pages 109–116, San Diego, California, March 2001.