Robust Sub-Sentential Alignment of Phrase-Structure Trees

Declan Groves
School of Computing
Dublin City University
Dublin 9, Ireland
dgroves@computing.dcu.ie

Mary Hearne
School of Computing
Dublin City University
Dublin 9, Ireland
mhearne@computing.dcu.ie

Andy Way
School of Computing
Dublin City University
Dublin 9, Ireland
away@computing.dcu.ie

Abstract

Data-Oriented Translation (DOT), based on Data-Oriented Parsing (DOP), is a language-independent MT engine which exploits parsed, aligned bitexts to produce very high quality translations. However, data acquisition constitutes a serious bottleneck as DOT requires parsed sentences aligned at both sentential and sub-structural levels. Manual sub-structural alignment is time-consuming, error-prone and requires considerable knowledge of both source and target languages and how they are related. Automating this process is essential in order to carry out the large-scale translation experiments necessary to assess the full potential of DOT.

We present a novel algorithm which automatically induces sub-structural alignments between context-free phrase structure trees in a fast and consistent fashion requiring little or no knowledge of the language pair. We present results from a number of experiments which indicate that our method provides a serious alternative to manual alignment.

1 Introduction

Approaches to Machine Translation (MT) using Data-Oriented Parsing (DOP: (Bod, 1998; Bod *et al.*, 2003)) require ⟨source,target⟩ tree fragments aligned at sentential and sub-sentential levels. In previous approaches to Data-Oriented Translation (DOT: (Poutsma, 2000; Hearne and Way, 2003)), such fragments were produced manually. This is time-consuming, error-prone, and requires considerable expertise of both source and target languages as well as how they are related. The obvious solution, therefore, is to automate the process of sub-sentential alignment. However, while there are many approaches to sentential alignment e.g. (Kay and Röscheisen, 1993; Gale & Church, 1993), no methods exist for aligning non-isomorphic phrase-structure (PS) tree fragments at sub-sentential level for use in MT. (Matsumoto *et al.*, 1993) align ⟨source,target⟩ dependency trees, with a view to resolve parsing ambiguities, but their approach cannot deal with complex or compound sentences. Other researchers (Imamura, 2001) also use phrase-alignment in parsing but in DOT the translation fragments are already in the form of parse-trees. (Eisner, 2003) outlines a computationally expensive structural manipulation tool which he has used for intra-lingual translation but has yet to apply to inter-lingual translation. (Gildea, 2003) performs tree-to-tree alignment, but treats it as part of a generative statistical translation model, rather than a seperate task. The method of (Ding *et al.*, 2003) can cope with a limited amount of non-isomorphism, but the algorithm is only suitable for use with dependency trees.

We develop a novel algorithm which automatically aligns translationally equivalent tree fragments in a fast and consistent fashion, and which requires little or no knowledge of the language pair. Our approach is similar to that of (Menezes and Richardson, 2003), who use a best-first approach to align dependency-type tree structures.

We conduct a number of experiments on the English-French section of the Xerox HomeCentre corpus. Using the manual alignment of (Hearne and Way, 2003) as a 'gold standard', we show that our algorithm identifies sub-structural translational equivalences with 73.7% precision and 67.84% recall. Furthermore, we replicate previous DOT experiments performed using manually aligned data. However, we use data aligned by our novel algorithm and evaluate the output translations. We demonstrate that while coverage decreases by 10%, the translations output are of comparable quality. These results indicate that our automatic alignment algorithm provides a serious alternative to manual alignment.

The remainder of this paper is organised as follows: in section 2, we discuss related research in more detail, while in section 3, we provide an overiew of DOT. We present our algorithm in section 4, and in section 5 describe the experiments conducted together with the results obtained. Finally, we conclude and provide avenues for further research.

2 Related Research

Several approaches to sub-structural alignment of tree representations have been proposed.

(Matsumoto *et al.*, 1993) and (Imamura, 2001) focus on using alignments to help resolve parsing ambiguities. As we wish to develop an alignment process for use in MT rather than parsing, this makes their approaches unsuitable for our use.

(Eisner, 2003) presents a tree-mapping method for use on dependency trees which he claims can be adapted for use with PS trees. He uses dynamic programming to break tree pairs into pairs of aligned elementary trees, similar to DOT. However, he aims to estimate a translation model from unaligned data, whereas we wish to align our data off-line. Currently, he has used his algorithm to perform intralingual translation but has yet to develop and apply real models to inter-lingual MT.

(Gildea, 2003) outlines an algorithm for use in syntax-based statistical models of MT, applying a statistical TSG with probabilities parameterized to generate the target tree conditioned on the structure of the source tree. His approach is unsuitable for DOT as it involves altering the shape of trees in order to impose isomorphism and the algorithm does not always generate a complete target tree structure. However, unlike (Gildea, 2003), we treat the problem of alignment as a seperate task rather than as part of a generative translation model.

(Ding *et al.*, 2003) and (Menezes and Richardson, 2003) also present approaches to the alignment of tree structures. Both deal with dependency structures rather than PS trees. (Ding *et al.*, 2003) outline an algorithm to extract word-level alignments using structural information taken from parallel dependency trees. They fix the nodes of tree pairs based on word alignments deduced statistically and then proceed by partitioning the tree into treelet pairs with the fixed nodes as their roots. Their algorithm relies on the fact that, in dependency trees, subtrees are headed by words rather than syntactic labels, making it unsuitable for our use.

(Menezes and Richardson, 2003) employ a best-first strategy and use a small alignment grammar to extract transfer mappings from bilingual corpora for use in translation. They use a bilingual dictionary and statistical techniques to supply translation pair candidates and to identify multi-word terms. Lexical correspondences are established using the lexicon of 98,000 translation pairs and a derivational morphology component to match other lexical items. Nodes are then aligned using these lexical correspondences along with structural information. Our algorithm uses a similar methodology. However, (Menezes and Richardson, 2003) use logical forms, which constitute a variation of dependency trees that normalize both the lexical and syntactic form of examples, whereas we align PS trees.

Although the methods outlined above have achieved promising results, only the approach of (Menezes and Richardson, 2003) seems relevant to our goal, even though they deal with abstract dependency-type structures rather than PS trees.

3 Data-Oriented Translation

Data-Oriented Translation (DOT) (Poutsma, 2000; Hearne and Way, 2003), which is based on Data-Oriented Parsing (DOP) (Bod, 1998; Bod *et al.*, 2003), comprises a context-rich, experience-based approach to translation, where new translations are derived with reference to grammatical analyses of previous translations. DOT exploits bilingual treebanks comprising linguistic representations of previously seen translation pairs, as well as explicit links which map the translational equivalences present within these pairs at sub-sentential level – an example of such a linked translation pair can be seen in Figure 1(a). Analyses and translations of the input are produced simultaneously by combining source and target language fragment pairs derived from the treebank trees.

3.1 Fragmentation

The tree fragment pairs used in Tree-DOT are called *subtree pairs* and are extracted from bilingual aligned treebank trees. The two decomposition operators, which are similar to those used in Tree-DOP but are refined to take the translational links into account, are as follows:

- the *root operator* which takes any pair of *linked* nodes in a tree pair to be the roots of a subtree pair and deletes all nodes except these new roots and all nodes dominated by them;
- the *frontier operator* which selects a (possibly empty) set of *linked* node pairs in the newly created subtree pairs, excluding the roots, and deletes all subtree pairs dominated by these nodes.

Allowing the *root* operator to select the root nodes of the original treebank tree pair and then the *frontier* operator to select an empty set of node pairs ensures that the original treebank tree pair is always included in the fragment base – in Figure 1, fragment (a) exactly matches the original treebank tree pair from which fragments (a) – (f) were derived. Fragments (b) and (f) were also derived by allowing the *frontier* operator to select the empty set; the *root* operation selected node pairs <A,N> and <NPadj,NPdet> respectively. Fragments (c), (d) and (e) were derived by selecting all further possible combinations of node pairs by *root* and *frontier*.

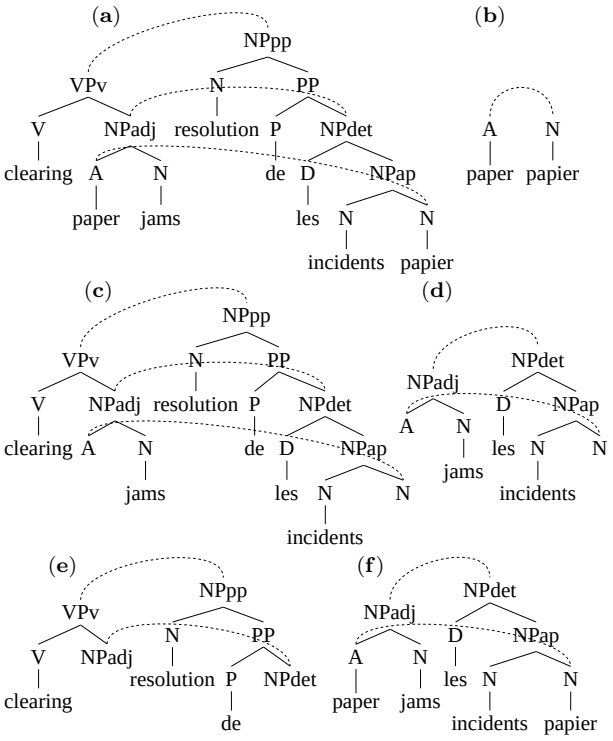

Figure 1: DOT fragments generated via *root* and *frontier*

3.2 Translation

The DOT composition operator is defined as follows. The composition of tree pairs $<s_1,t_1>$ and $<s_2,t_2>$ ($<s_1,t_1> \circ <s_2,t_2>$) is only possible if

- the leftmost non-terminal frontier node of s_1 is of the same syntactic category (e.g. S, NP, VP) as the root node of s_2, and
- the leftmost non-terminal frontier node of s_1's *linked counterpart* in t_1 is of the same syntactic category as the root node of t_2.

The resulting tree pair consists of a copy of s_1 where s_2 has been inserted at the leftmost frontier node and a copy of t_1 where t_2 has been inserted at the node linked to s_1's leftmost frontier node, as illustrated in Figure 2.

The DOT probability of a translation derivation is the joint probability of choosing each of the subtree pairs involved in that derivation. The probability of selecting a subtree pair is its number of occurrences in the corpus divided by the number of pairs in the corpus with the same root nodes as it:

$$P(< e_s, e_t >) = \frac{|<e_s,e_t>|}{\sum_{<u_s,u_t>:r(<u_s,u_t>)=r(<e_s,e_t>)} |<u_s,u_t>|}$$

The probability of a derivation in DOT is the product of the probabilities of the subtree pairs involved in building that derivation. Thus, the probability of derivation $<s_1,t_1> \circ ... \circ <s_n,t_n>$ is given by

$$P(< s_1, t_1 > \circ ... \circ < s_n, t_n >) = \prod_i P(< s_i, t_i >)$$

Again, a translation can be generated by many different derivations, so the probability of a translation $w_s \longleftrightarrow w_t$ is the sum of the probabilities of its derivations:

$$P(< w_s, w_t >) = \sum_{<t_{s_i},t_{t_i}> yields <w_s,w_t>} P(< t_{s_i}, t_{t_i} >)$$

Selection of the most probable translation *via* Monte Carlo sampling involves taking a random sample of derivations and outputting the most frequently occurring translation in the sample.

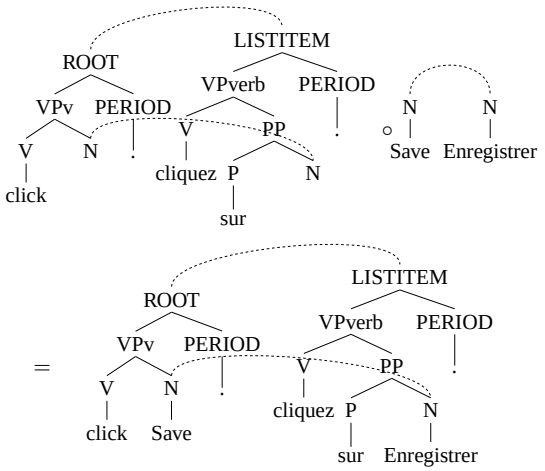

Figure 2: The DOT composition operation

4 Our Algorithm

The operation of a DOT system is dependent on the availability of bilingual treebanks aligned at sentential and sub-sentential level. Our novel algorithm attempts to fully automate sub-sentential alignment using an approach inspired by that of (Menezes and Richardson, 2003). The algorithm takes as input a pair of ⟨source,target⟩ PS trees and outputs a mapping between the nodes of the tree pair.

As with the majority of previous approaches, the algorithm starts by finding lexical correspondences between the source and target trees. Our lexicon is built automatically using a previously developed word aligner based on the k-vec aligner as outlined by (Fung & Church, 1994). This lexical aligner uses a combination of automatically extracted cognate information, mutual information and probabilistic measures to obtain one-to-one lexical correspondences between the source and target strings. During lexical alignment, function words are excluded because, as they are the most common words in a

language, they tend to co-occur frequently with the content words they precede. This can lead to the incorrect alignment of content words with function words.

The algorithm then proceeds from the aligned lexical terminal nodes in a bottom-up fashion, using a mixture of node label matching and structural information to perform language-independent linking between all ⟨source,target⟩ node pairs within the trees. As with (Menezes and Richardson, 2003), it uses a best-first approach. After each step, new linked node pairs are added to the current list of linked nodes. The links made between the nodes are fixed, thus restricting the freedom of alignment for the remaining unaligned nodes in the tree pair. The methods of the algorithm are applied to each new linked node pair in turn until no new node pairs can be added. The algorithm consists of five main methods which are performed on each linked node pair in the list:

Verb + Object Align (Figure 3): We have a linked source-target node pair ⟨*s*,*t*⟩. *s* and *t* are both verbs, are the leftmost children in their respective trees, both have VP parent nodes and they have the same number of siblings which have similar syntactic labels. We align the corresponding siblings of *s* and *t*. This aligns the objects of the source verb with the equivalent objects of the target verb. We also align the parents of *s* and *t*.

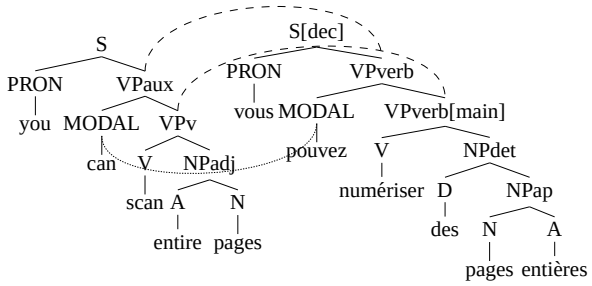

Figure 3: Verb + Object Align: *the dashed lines represent the links made by Verb + Object Align when the current linked node pair is ⟨MODAL,MODAL⟩.*

Parent Align (Figure 4): We have a current linked source-target node pair ⟨*s*,*t*⟩ with unlinked parents par_s and par_t respectively. All the sister nodes of *s* are aligned with sister nodes of *t*. We link par_s and par_t. If *s* and *t* each have one unlinked sister, but the remaining sisters of *s* are aligned with sister nodes of *t*, link the unlinked sisters and link par_s with par_t.

NP/VP Align (Figure 5): We have a linked source-target node pair ⟨*s*,*t*⟩ and *s* and *t* are both nouns. Traverse up the source tree to find the topmost NP node np_s dominating *s* and traverse up the target tree to find the topmost target NP node np_t dominating *t*. We link np_s and np_t. We then traverse down from np_s and np_t to the leftmost leaf nodes (l_s and l_t) in the source and target subtrees rooted at np_s and np_t. If l_s and l_t have similar labels, we link them. This helps to preserve the scope of noun-phrase modifiers. If *s* and *t* are both verbs, we perform a similar method, this time linking the topmost VP nodes in the source and target trees.

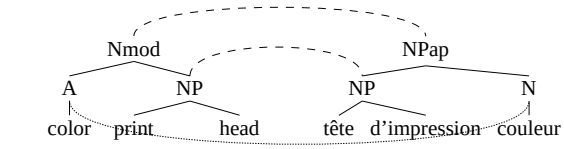

Figure 4: Parent Align: *The dashed lines are the links made by Parent Align, when ⟨color,couleur⟩ is the current linked node pair.*

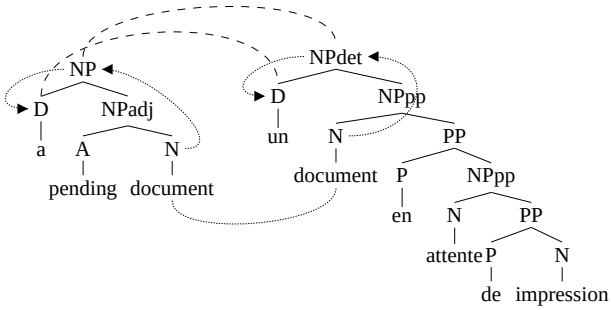

Figure 5: NP Align: *the dashed lines represent the links made by NP Align when the current linked node pair is ⟨N,N⟩.*

Child Align (Figure 6): This method is similar to that of Parent Align. We have a current linked source-target node pair ⟨*s*,*t*⟩. Each node has the same number of children and these children have similar node labels. We link their corresponding children.

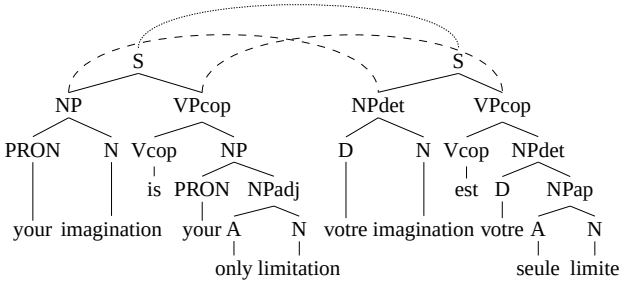

Figure 6: Child Align: *the dashed lines represent the links made by Child Align when the current linked node pair is ⟨S,S⟩.*

Subtree Align: We have a linked source-target node pair ⟨*s*,*t*⟩. If the subtrees rooted at *s* and at *t* are fully isomorphic, we link the corresponding nodes within the subtrees. This accounts for the fact that trees may not be completely isomorphic from their roots but may be isomorphic at subtree level.[1]

[1]Originally we used a method *isomophic* which checked for

Once lexical correspondences have been established, the methods outlined above use structural information to align the ⟨source,target⟩ nodes. The comparison of ⟨source,target⟩ node labels during alignment ensures that sub-structures with corresponding syntactic categories are aligned. If the algorithm fails to find any alignments between the source and target tree pairs, due to the absence of initial lexical correspondences, we align the ⟨source,target⟩ root nodes.

5 Experiments and results

Previous DOT experiments (Hearne and Way, 2003) were carried out on a subset of the HomeCentre corpus consisting of 605 English-French sentence pairs from Xerox documentation parsed into LFG c(onstituent)- and f(unctional)-structure representations and aligned at sentence level. This bilingual treebank constitutes a linguistically complex fragment base containing many 'hard' translation examples, including cases of nominalisations, passivisation, complex coordination and combinations thereof. Accordingly, the corpus would appear to present a challenge to any MT system.

The insertion of the links denoting translational equivalence for the set of tree pairs used in the previous experiments was performed manually. We have applied our automatic sub-structural alignment algorithm to this same set of 605 tree pairs and evaluated performance using two distinct methods. Firstly, we used the manual alignments as a 'gold standard' against which we evaluated the output of the alignment algorithm in terms of precision, recall and f-score. The results of this evaluation are presented in Section 5.1. Secondly, we repeated the DOT experiments described in (Hearne and Way, 2003) using the automatically generated alignments in place of those determined manually. We evaluated the output translations in terms of IBM Bleu scores, precision, recall and f-score and present these results in Section 5.2.

5.1 Evaluation of alignment quality

Using the manually aligned tree pairs as a 'gold standard', we evaluated the performance of each of the five methods which constitute the alignment algorithm both individually and in combination. These evaluations are summarised in Figures 7 and 8 respectively.

The alignment process is always initialised by finding word correspondences between the source and target trees, meaning that lexical alignment is carried out regardless of which other method or combination of methods is included. The low rate of recall achieved by the lexical alignment process of 0.3057, shown in Figure 7, can be largely attributed to the fact that it does not align function words. We achieve high precision relative to recall – as is generally preferred for automatic procedures – indicating that the alignments induced are more likely to be 'partial' than incorrect.

When evaluated individually, the *Parent Align* method performs best, achieving an f-score of 0.5978. Overall, the highest f-score of 0.7064 is achieved by using all methods, including the additional *subtree* method, in combination.

	PRECISION	RECALL	F-SCORE
Lex	0.6800	0.3057	0.4212
Par	**0.7471**	**0.4983**	**0.5978**
NP/VP	0.7206	0.4879	0.5819
Child	0.7045	0.3856	0.4984
Verb + Object	0.6843	0.3191	0.4352

Figure 7: Individual evaluation of alignment methods

	PRECISION	RECALL	F-SCORE
Par + Child	0.7525	0.5588	0.6414
Par + NP/VP	0.7373	0.6106	0.6680
Par + Child + NP/VP	0.7411	0.6587	0.6974
All	**0.7430**	0.6686	0.7039
All + Subtree	0.7370	**0.6784**	**0.7064**

Figure 8: Evaluation of combined alignment methods

5.2 Evaluation of translation quality

In order to evaluate the impact of using automatically generated alignments on translation quality, we repeated the DOT experiments described in (Hearne and Way, 2003) using these alignments in place of manually determined translational equivalences.

In order to ensure that differences in the results achieved could be attributed solely to the different sub-structural alignments imposed, we used precisely the same 8 training/test set splits as before, where each training set contained 545 parsed sentence pairs, each test set 60 sentences, and all words occurring in the source side of the test set also occurred in the source side of the training set (but not necessarily with the same lexical category). As before, all translations carried out were from English into French and the number of samples taken during the disambiguation process was limited to 5000.

Due to constraints on time and memory, data-oriented language processing applications generally limit the size of the fragment base by exclud-

isomorphism from the roots downwards, assuming a root-root correspondence. However, this significantly decreased the performance of the aligner.

	Bleu/Auto	Bleu/Man	F-Score/Aut.	F-Score/Man
LD1	0.0605	**0.2627**	0.3558	**0.5506**
LD2	0.1902	**0.3018**	0.4867	**0.5870**
LD3	0.1983	**0.3235**	0.4957	**0.6045**
LD4	0.214	**0.3235**	0.5042	**0.6069**

Figure 9: Evaluation over all translations

	Bleu/Auto	Bleu/Man	F-Score/Auto	F-Score/Man
LD1	0.6118	**0.6591**	0.7900	**0.8090**
LD2	**0.7519**	0.7144	**0.8751**	0.8446
LD3	**0.7790**	0.7610	**0.8887**	0.8688
LD4	**0.7940**	0.7611	**0.8930**	0.8736

Figure 10: Evaluation over translations produced

	Bleu/Auto	Bleu/Man	F-Score/Auto	F-Score/Man
LD1	0.5945	**0.6363**	0.7918	**0.7989**
LD2	0.7293	**0.7382**	**0.8823**	0.8629
LD3	0.7700	**0.7930**	**0.8938**	0.8913
LD4	0.7815	**0.7940**	**0.8964**	0.8933

Figure 11: Evaluation of sentences translated by both alignment methods

ing larger fragments. In these experiments, we increased the size of the fragment base incrementally by initially allowing only fragments of link depth (LD) 1 and then including those of LD 2, 3 and 4. [2]

We evaluated the output translations in terms of IBM Bleu scores using the NIST MT Evaluation Toolkit[3] and in terms of precision, recall and f-score using the NYU General Text Matcher.[4] We summarise our results and reproduce and extend those of (Hearne and Way, 2003)[5] in Figures 9, 10 and 11.

Results over the full set of output translations, summarised in Figure 9, show that using the manually linked fragment base results in significantly better overall performance at all link depths (LD1 - LD4) than using the automatic alignments. However, both metrics used assign score 0 in all instances where no translation was output by the system. The comparatively poor scores achieved using the automatically induced alignments reflect the fact that these alignments give poorer coverage at all depths than those determined manually (47.71% vs. 66.46% at depth 1, 56.39% vs. 67.92% at depths 2 - 4).

The results in Figure 10 include scores only where a translation was produced. Here, translations produced using manual alignments score better only at LD 1; better performance is achieved at LD 2 - 4 using the automatically linked fragment base. Again, this may – at least in part – be an issue of coverage: many of the sentences for which only the manually aligned fragment base produces translations are translationally complex and, therefore, more likely to be only partially correct and achieve poor scores.

Finally, we determined the subset of sentences for which translations were produced both when the manually aligned fragment bases were used and when the automatically linked ones were used. Figure 11 summarises the results achieved when evaluating only these translations. In terms of Bleu scores, translations produced using manual alignments score slightly better at all depths. However, as link depth increases the gap narrows consistently and at depth 4 the difference in scores is reduced to just 0.0125. In terms of f-scores, the translations produced using automatic alignments actually score better than those produced using manual alignments at depths 2 - 4.

5.3 Discussion

Our first evaluation method (Section 5.1) is, perhaps, the obvious one to use when evaluating alignment performance. However, the results of this evaluation, which show best f-scores of 70%, provide no insight into the effect using these alignments has on translation accuracy. Evaluating these alignments in context – by using them in the DOT system for which they were intended – gives us a true picture of their worth. Crucially, in Section 5.2 we showed that using automatic rather than manual alignments results in translations of extremely high quality, comparable to those produced using manual alignments.

In many cases, translations produced using automatic alignments contain fewer errors involving local syntactic phenomena than those produced using manual alignment. This suggests that, as links between function words are infrequent in the automatic alignments, we achieve better modelling of phenomena such as determiner-noun agreement because the determiner fragments do not generally occur without context. For example, there are relatively few instances of 'D→the' aligned with 'D→le/la/l'/les' in the automatic alignment compared to the manual alignment.

[2] The link depth of a fragment pair is defined as greatest number of steps taken *which depart from a linked node* to get from the root node to any frontier nodes (Hearne and Way, 2003).

[3] http://www.nist.gov/speech/tests/mt/mt2001/resource/

[4] http://nlp.cs.nyu.edu/GTM/

[5] The Bleu scores shown here differ from those published in (Hearne and Way, 2003) as a result of recent modifications to the NIST MT Evaluation Kit.

On the other hand, we achieve 10% less coverage when translating using the automatic alignments. The automatic alignments are less likely to identify non-local phenomena such as long-distance dependencies. Consequently, the sentences only translated when using manual alignments are generally longer and more complex than those translated by both. While a degree of trade-off between coverage and accuracy is to be expected, we would like to increase coverage while maintaining or improving translation quality. Improvements to lexical alignment should prove valuable in this regard. While we expect translation quality to improve as depth increases, experiments using the automatical alignment show disproportionately poor performance at depth 1. The majority of links in the depth 1 fragment base are inserted using the lexical aligner, indicating that these are less than satisfactory. We expect improvements to the lexical aligner to significantly improve the overall performance of the alignment algorithm and, consequently, the quality of the translations produced. Lexical alignment is crucial in identifying complex phenomena such as long distance dependencies. Using machine-readable bilingual dictionaries or, alternatively, manually established word-alignments to intiate the automatic substructural alignment algorithm may provide more accurate results.

6 Conclusions and future work

We have presented an automatic algorithm which aligns bilingual context-free phrase-structure trees at sub-structural level and applied this algorithm to a subset of the English-French section of the HomeCentre corpus. We have outlined detailed evaluations of our algorithm. They show that while translation coverage was 10% lower using the automatically aligned data, the quality of the translations produced is comparable to the quality of those produced using manual alignments. While DOT systems produce very high quality translations in reasonable time, resource acquisition remains an issue. Manual sub-structural alignment is time-consuming, error-prone and requires considerable linguistic expertise. Our alignment method, on the other hand, is efficient, consistent and language-independent, constituting a viable alternative to manual sub-structural alignment; thus solving the data acquisition problem.

We intend to apply our automatic alignment methodology to the full English-French section of the HomeCentre corpus, as well as the English-German and French-German sections, and perform experiments to further validate the the language-independent nature of both our alignment algorithm and the data-oriented approach to translation. We also plan to automatically parse existing bitexts, thus creating further resources for use with our DOT system and, together with our aligner, enabling much larger-scale DOT-based translation experiments than have been performed to date.

7 Aknowledgements

The work presented in this paper is partly supported by an IRCSET [6] PhD Fellowship Award.

References

Rens Bod. 1998. *Beyond Grammar: An Experience-Based Theory of Language*. CSLI, Stanford, CA.

Rens Bod, Remko Scha and Khalil Sima'an. (eds.) 2003. *Data-Oriented Parsing*. CSLI, Stanford CA.

Yuan Ding, Dan Gildea and Martha Palmer. 2003. An Algorithm for Word-Level Alignment of Parallel Dependency Trees. *MT Summit IX*. New Orleans, LO., pp.95–101.

Jason Eisner. 2003. Learning Non-Isomorphic Tree Mappings for Machine Translation. In *Proceedings of the 41st COLING*, Sapporo, Japan.

Pascale Fung & Ken W. Church. 1994. K-vec: A New Approach for Aligning Parallel Texts. In *Proceedings of COLING 94*, Kyoto, Japan, pp.1096-1102.

William A. Gale & Ken W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics* **19**(1):75–102.

Daniel Gildea. 2003. Loosely Tree-Based Alignment for Machine Translation. In *Proceedings of the 41st ACL*. Sapporo, Japan, pp.80–87.

Mary Hearne and Andy Way. 2003. Seeing the Wood for the Trees: Data-Oriented Translation. *MT Summit IX*. New Orleans, LO., pp.165–172.

Kenji Imamura. 2001. Hierarchical Phrase Alignment Harmonized With Parsing. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*. Tokyo, Japan, pp.377–384.

Martin Kay and Martin Röscheisen. 1993. Text-translation alignment. *Computational Linguistics* **19**(1):121–142.

Yuji Matsumoto, Ishimoto Hiroyuki and Takehito Utsuro. 1993. Structural Matching of Parallel Texts. In *Proceedings of the 31st ACL*. Columbus, OH., pp.23–30.

Arul Menezes and Stephen D. Richardson. 2003. A Best-First Alignment Algorithm for Automatic Extraction of Transfer Mappings from Bilingual Corpora. In M. Carl & A. Way (eds.) *Recent Advances in Example-Based Machine Translation*. Kluwer Academic Publishers, Dordrecht, The Netherlands, pp.421–442.

Arjen Poutsma. 2000. Data-Oriented Translation. In *18th COLING*, Saarbrücken, Germany, pp.635–641.

[6] http://www.ircset.ie