Computing translation units and quantifying parallelism in parallel dependency treebanks

Matthias Buch-Kromann
ISV Computational Linguistics Group
Copenhagen Business School
mbk.isv@cbs.dk

Abstract

The linguistic quality of a parallel treebank depends crucially on the parallelism between the source and target language annotations. We propose a linguistic notion of translation units and a quantitative measure of parallelism for parallel dependency treebanks, and demonstrate how the proposed translation units and parallelism measure can be used to compute transfer rules, spot annotation errors, and compare different annotation schemes with respect to each other. The proposal is evaluated on the 100,000 word Copenhagen Danish-English Dependency Treebank.

1 Introduction

Parallel treebanks are increasingly seen as a valuable resource for many different tasks, including machine translation, word alignment, translation studies and contrastive linguistics (Čmejrek et al., 2004; Cyrus, 2006; Hansen-Schirra et al., 2006). However, the usefulness of a parallel treebank for these purposes is directly correlated with the degree of syntactic parallelism in the treebank. Some non-parallelism is inevitable because two languages always differ with respect to their syntactic structure. But non-parallelism can also be the result of differences in the linguistic analyses of the source text and target text, eg, with respect to whether noun phrases are headed by nouns or determiners, whether conjunctions are headed by the first conjunct or the coordinator, whether prepositions are analyzed as heads or adjuncts in prepositional phrases, etc.

In this paper, we focus on parallel dependency treebanks that consist of source texts and translations annotated with dependency analyses and word-alignments. These requirements are directly satisfied by the analytical layer of the Prague Czech-English Dependency Treebank (Čmejrek et al., 2004) and by the dependency layer of the Copenhagen Danish-English Dependency Treebank (Buch-Kromann et al., 2007). The requirements are also indirectly satisifed by parallel treebanks with a constituent layer and a word alignment, eg (Han et al., 2002; Cyrus, 2006; Hansen-Schirra et al., 2006; Samuelsson and Volk, 2006), since it is possible to transform constituent structures into dependency structures — a procedure used in the CoNLL shared tasks in 2006 and 2007 (Buchholz and Marsi, 2006). Finally, it is worth pointing out that the requirements are also met by any corpus equipped with two different dependency annotations since a text is always trivially word-aligned with itself. The methods proposed in the paper therefore apply to a wide range of parallel treebanks, as well as to comparing two monolingual treebank annotations with each other.

The paper is structured as follows. In section 2, we define our notions of word alignments and dependencies. In section 3, we define our notion of translation units and state an algorithm for computing the translation units in a parallel dependency treebank. Finally, in sections 4, 5 and 6, we demonstrate how translation units can be used to compute transfer rules, quantify parallelism, spot annotation errors, and compare monolingual and bilingual annotation schemes with respect to each other.

Proceedings of the Linguistic Annotation Workshop, pages 69–76,
Prague, June 2007.

Complement roles		Adjunct roles	
aobj	adjectival object	**appa**	parenthetical apposition
avobj	adverbial object	**appr**	restrictive apposition
conj	conjunct of coordinator	**coord**	coordination
dobj	direct object	**list**	unanalyzed sequence
expl	expletive subject	**mod**	modifier
iobj	indirect object	**modo**	dobj-oriented modifier
lobj	locative-directional obj.	**modp**	parenthetical modifier
nobj	nominal object	**modr**	restrictive modifier
numa	additive numeral	**mods**	subject-oriented mod.
numm	multiplicative numeral	**name**	additional proper name
part	verbal particle	**namef**	additional first name
pobj	prepositional object	**namel**	additional last name
possd	possessed in genitives	**pnct**	punctuation modifier
pred	subject/object predicate	**rel**	relative clause
qobj	quotation object	**title**	title of person
subj	subject	**xpl**	explification (colon)
vobj	verbal object		

Figure 1: The main dependency roles in the dependency framework Discontinuous Grammar.

2 Word alignments and dependencies

In our linguistic analyses, we will assume that a word alignment $W \leftrightarrow W'$ encodes a translational correspondence between the word clusters W and W' in the source text and target text, ie, the word alignment expresses the human intuition that the subset W of words in the source text corresponds roughly in meaning or function to the subset W' of words in the target text. The translations may contain additions or deletions, ie, W and W' may be empty.

We also assume that a dependency edge $g \xrightarrow{r} d$ encodes a complement or adjunct relation between a word g (the *governor*) and a complement or adjunct phrase headed by the word d (the *dependent*), where the edge label r specifies the complement or adjunct dependency role.[1] As an illustration of how complement and adjunct relations can be encoded by means of dependency roles, the most important dependency roles used in the dependency framework Discontinuous Grammar (Buch-Kromann, 2006) are shown in Figure 1. Finally, we will assume that the dependencies form a tree (or a forest). The tree may be non-projective, ie, it may contain crossing branches (technically, a dependency $g \xrightarrow{r} d$ is projective if and only if g is a transitive governor of all the words between g and d).

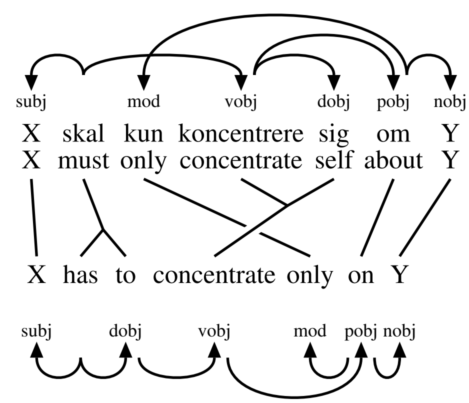

Figure 2: Parallel dependency treebank analysis with word alignment and two monolingual dependency analyses.

Figure 2 shows an example of this kind of analysis, based on the annotation conventions used in Discontinuous Grammar and the associated Copenhagen Danish-English Dependency Treebank (Buch-Kromann et al., 2007). In the example, word alignments are indicated by lines connecting Danish word clusters with English word clusters, and dependencies are indicated by means of arrows that point from the governor to the dependent, with the dependency role written at the arrow tip. For example, the Danish word cluster "koncentrere sig" ("concentrate self") has been aligned with the English word "concentrate", and the English phrase

[1]Following standard dependency theoretic assumptions, we will assume the following differences between complement and adjunct relations: (a) complements are lexically licensed by their governor, whereas adjuncts license their adjunct governor; (b) in the functor-argument structure, complements act as arguments of their governor, whereas adjuncts act as modifiers of their governor; (c) a governor can have several adjuncts with the same adjunct role, whereas no two complements of the same governor can have the same complement role.

headed by "on" is analyzed as a prepositional object of the verb "concentrate." In the Danish dependency analysis, the dependency between the adverbial "kun" ("only") and its prepositional governor "om" ("about") is non-projective because "om" does not dominate the words "koncentrere" ("concentrate") and "selv" ("self").

Dependency analyses differ from phrase-structure analyses in that phrases are a derived notion: in a dependency tree, each word has a derived phrase that consists of all the words that can be reached from the word by following the arrows. For example, the English word "concentrate" heads the phrase "concentrate only on Y," and the Danish word "om" heads the discontinuous phrase "kun … om Y."

If a parallel dependency analysis is well-formed, in a sense to be made clear in the following section, each alignment edge corresponds to what we will call a translation unit. Intuitively, given an aligment edge $W \leftrightarrow W'$, we can create the corresponding translation unit by taking the source and target subtrees headed by the words in W and W', deleting all parallel adjuncts of $W \leftrightarrow W'$, and replacing all remaining parallel dependents of $W \leftrightarrow W'$ with variables $x_1, \ldots, x_n$ and $x'_1, \ldots, x'_n$. The resulting translation unit will be denoted by $T(x_1, \ldots, x_n) \leftrightarrow T'(x'_1, \ldots, x'_n)$, where T and T' denote the source and target dependency trees in the translation unit. For convenience, we will sometimes use vector notation and write $T(\mathbf{x}) \leftrightarrow T'(\mathbf{x}')$ instead of $T(x_1, \ldots, x_n) \leftrightarrow T'(x'_1, \ldots, x'_n)$. Dependencies are usually defined as relations between words, but by an abuse of terminology, we will say that a word d is a *dependent* of an alignment edge $W \leftrightarrow W'$ provided d is a dependent of some word in $W \cup W'$ and d is not itself contained in $W \cup W'$.

Figure 3 shows the six translation units that can be derived from the parallel dependency analysis in Figure 2, by means of the procedure outlined above. Each translation unit can be interpreted as a bidirectional translation rules: eg, the second translation unit in Figure 3 can be interpreted as a translation rule stating that a Danish dependency tree with terminals "x_1 skal x_2" can be translated into an English dependency tree with terminals "x'_1 has to x'_2" where the English phrases x'_1, x'_2 are translations of the Danish phrases x_1, x_2, and vice versa.

In the following section, we will go deeper into the definition and interpretation of these rules. In particular, unlike the essentially context-free translation rules used in frameworks such as (Quirk et al., 2005; Ding, 2006; Chiang, 2007), we will not assume that the words in the translation rules are ordered, and that the translation rules can only be used in a way that leads to projective dependency trees.

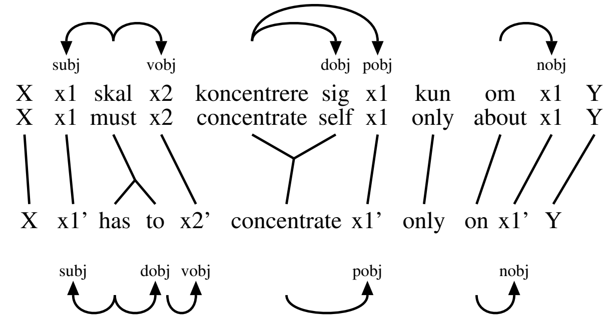

Figure 3: The six translation units derived from the parallel dependency analysis in Figure 2.

3 Translation units within a simple dependency-based translation model

In many parallel treebanks, word alignments and syntactic annotations are created independently of each other, and there is therefore no guarantee that the word or phrase alignments coincide with any meaningful notion of translation units. To rectify this problem, we need to define a notion of translation units that links the word alignments and the source and target dependency analysis in a meaningful way, and we need to specify a procedure for constructing a meaningful set of word alignments from the actual treebank annotation.

Statistical machine translation models often embody an explicit notion of translation units. However, many of these models are not applicable to parallel treebanks because they assume translation units where either the source text, the target text or both are represented as word sequences without any syntactic structure (Galley et al., 2004; Marcu et al., 2006; Koehn et al., 2003). Other SMT models assume translation units where the source and target language annotation is based on either context-free grammar (Yamada and Knight, 2001; Chiang, 2007) or context-free dependency grammar (Quirk et al., 2005; Ding, 2006). However, since nonprojectivity is not directly compatible with context-free grammar, and parallel dependency treebanks tend to encode non-projective dependencies directly, the context-free SMT models are not directly applicable to parallel dependency treebanks in general. But the context-free SMT models are an important inspiration for the simple dependency-based translation model and notion of translation units that we will present below.

In our translation model, we will for simplicity assume that both the source dependency analysis and the target dependency analysis are unordered trees, ie, dependency transfer and word ordering are modelled as two separate processes. In this paper, we only look at the dependency transfer and ignore the word ordering, as well as the probabilistic modelling of the rules for transfer and word ordering. There are three kinds of translation rules in the model:

A. Complement rules have the form $T(\mathbf{x}) \leftrightarrow T'(\mathbf{x}')$ where $T(\mathbf{x})$ is a source dependency tree with variables $\mathbf{x} = (x_1, \ldots, x_n)$, $T'(\mathbf{x}')$ is a target dependency tree with variables $\mathbf{x}' = (x'_1, \ldots, x'_n)$, the words in T are aligned to the words in T', and the variables x_k, x'_k denote parallel source and target subtrees. The rule states that a source tree $T(\mathbf{x})$ can be transferred into a target tree $T'(\mathbf{x}')$ by transferring the source subtrees in $\mathbf{x}$ into the target subtrees in $\mathbf{x}'$.

B. Adjunct rules have the form $(x \xleftarrow{a} T(\mathbf{y})) \leftrightarrow (x' \xleftarrow{a'} T'(\mathbf{y}'))$ where $T(\mathbf{y})$ is a source dependency tree, $T'(\mathbf{y}')$ is a target dependency tree, and x, x' are variables that denote parallel adjunct subtrees with adjunct roles a, a', respectively. The rule states that given a translation unit $T(\mathbf{y}) \leftrightarrow T(\mathbf{y}')$, an a-adjunct of any word in T can be translated into an a'-adjunct of any word in T'.[2]

C. Addition/deletion rules have the form $T(\mathbf{y}) \leftrightarrow (x' \xleftarrow{a'} T'(\mathbf{y}'))$ and $(x \xleftarrow{a} T(\mathbf{y})) \leftrightarrow T'(\mathbf{y}')$ where x, x' are variables that denote adjunct subtrees, and a, a' are adjunct relations. The addition rule states that an adjunct subtree x' can be introduced into the target tree T' in a translation unit $T(\mathbf{y}) \leftrightarrow T(\mathbf{y}')$ without any corresponding adjunct in the source tree T. Similarly, the deletion rule states that the adjunct subtree x in the source tree T does not have to correspond to any adjunct in the target tree T'.[3]

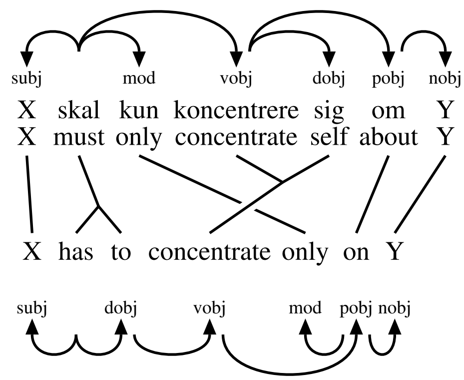

Figure 4: Parallel dependency analysis that is incompatible with our translation model.

The translation model places severe restrictions on the parallel dependency annotations. For example, the annotation in Figure 4 is incompatible with our proposed translation model with respect to the adjunct "only', since "only" attaches to the verb "skal/must" in the Danish analysis, but attaches to the preposition "on" in the English analysis — ie, it does not satisfy a requirement that follows implicitly from the adjunct rule: that corresponding source and target adjunct governors must belong to the same translation unit. In our example, there are two ways of rectifying the problem: we can (a) correct the dependency analysis as shown in Figure 2, or (b) correct the word alignment as shown in Figure 5.

It can be shown that our translation model translates into the following four requirements on parallel analyses — ie, the requirements are necessary and sufficient for ensuring that the linguistic annotations are compatible with our translation model. In the following, two words are said to be *coaligned* if they belong to the same alignment edge. A dependency edge $d \xleftarrow{r} g$ is called *internal* if d and g are coaligned, and *external* otherwise. A word w is called *singular* if it fails to be coaligned with at least one word in the other language.

Requirement I. The internal dependencies within a translation unit must form two connected trees. Ie,

[2]In the form stated here, adjunct rules obviously overgenerate because they do not place any restrictions on the words in T' that the target adjunct can attach to. In a full-fledged translation model, the adjunct rules must be augmented with a probabilistic model that can keep track of these restrictions.

[3]As with adjunct rules, addition/deletion rules obviously overgenerate, and must be augmented with probabilistic models that keep track of the precise characteristics of the adjunct subtrees that are added to or deleted from the parallel analyses.

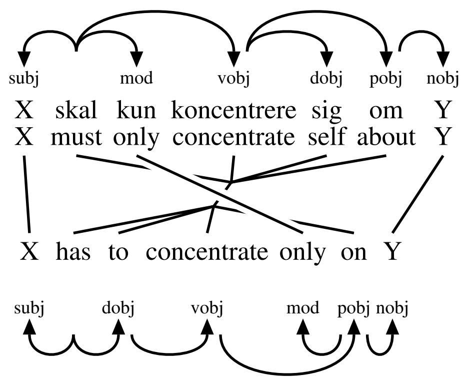

Figure 5: Making the analysis from Figure 4 compatible with our translation model by changing the alignment edges.

in an alignment $W \leftrightarrow W'$, the internal dependencies within W must form a connected source tree, and similarly for W'.

Requirement II. The external dependencies between translation units must form an acyclic graph. Ie, in an alignment $W \leftrightarrow W'$, no word $w \in W$ can be coaligned with an external transitive dependent of any word in W', and vice versa.

Requirement III. Parallel external governors must be aligned to each other. Ie, if two nodes n, n' are coaligned with external governor edges $n \xleftarrow{r} g$ and $n' \xleftarrow{r'} g'$, then g and g' must be coaligned.

Requirement IV. The graph contains no singular external complements. If the source word c is a complement of governor g and c is not coaligned to any target word, then c and g must be coaligned to each other; and similarly for target complements.

A graph that satisfies all four requirements is said to be *well-formed* with respect to our translation model. It can be shown that we can always transform an ill-formed graph G into a well-formed graph G' by merging alignment edges; G' is then called a *reduction* of G, and a reduction with a minimal number of mergings is called a *minimal reduction* of G. In a well-formed graph, we will refer to an alignment edge and its associated source and target dependency tree as a *translation unit*.

It can be shown that minimal reductions can be computed by means of the algorithm shown in Figure 6.[4] The body of the for-loop in Figure 6 ensures

procedure minimal-reduction(graph G)
    merge each alignment edge in G with itself
      (ie, ensure int. connectedness & ext. acyclicity)
    for *each* $W \leftrightarrow W'$ *in bottom-up order* **do**
        merge $W \leftrightarrow W'$ *with all of its external singular complements*
        merge all external governors of $W \leftrightarrow W'$
    return *the modified graph G*

Figure 6: Algorithm for computing the minimal reduction of a graph G.

Requirements III (coaligned external governors) and IV (no singular complements), and the merging operation is designed so that it ensures Requirements I (internal connectedness) and II (acyclicity).[5]

The ill-formed analysis in Figure 4 has the minimal reduction shown in Figure 2, whereas the analyses in Figure 2 and 5 are well-formed, ie, they are their own minimal reductions. In the remainder of the paper, we will describe how minimal reductions and translation units can be used for extracting transfer rules, detecting annotation errors, and comparing different annotation schemes with each other.

4 Extracting transfer rules and quantifying parallelism

The complement, adjunct, and addition/deletion rules in our simple dependency transfer model can be read off directly from the minimal reductions. Figure 7 shows the three complement rules induced from Figure 4 via the minimal reduction in Figure 5. Figure 8 (repeated from Figure 3) shows the six complement rules induced from the alternative analysis in Figure 2.

We have tested the extraction procedure on a large scale by applying it to the 100,000 word Copenhagen Danish-English Dependency Treebank (Buch-Kromann et al., 2007). Figure 9 shows the percentage of translation units with size at least n

[4] In the algorithm, G is viewed as a directed graph that contains the source and target dependencies, and alignment edges $W \leftrightarrow W'$ are viewed as short-hands for the set of all bidirectional edges that link two distinct nodes in $W \cup W'$.

[5] The merging operation performs three steps: (a) replace two alignment edges $W_1 \leftrightarrow W_1'$ and $W_2 \leftrightarrow W_2'$ with their union $W \leftrightarrow W'$ where $W = W_1 \cup W_2$ and $W' = W_1' \cup W_2'$; (b) merge $W \leftrightarrow W'$ with the smallest set of nodes that turns W and W' into connected dependency trees; (c) merge $W \leftrightarrow W'$ with all nodes on cycles that involve at least one node from $W \leftrightarrow W'$.

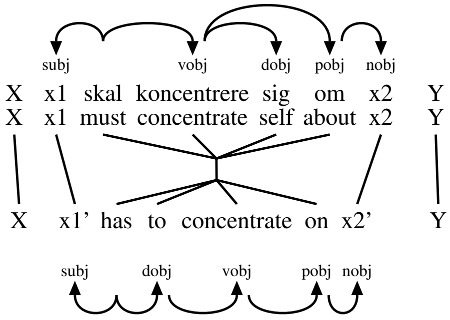

Figure 7: The three complement rules induced from Figure 4 via the minimal reduction in Figure 5.

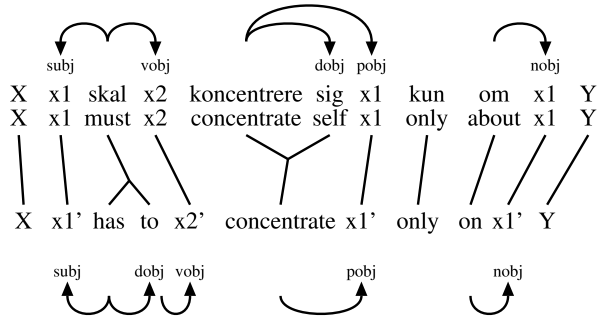

Figure 8: The six complement rules induced from the minimal reduction in Figure 2 (repeated from Figure 3).

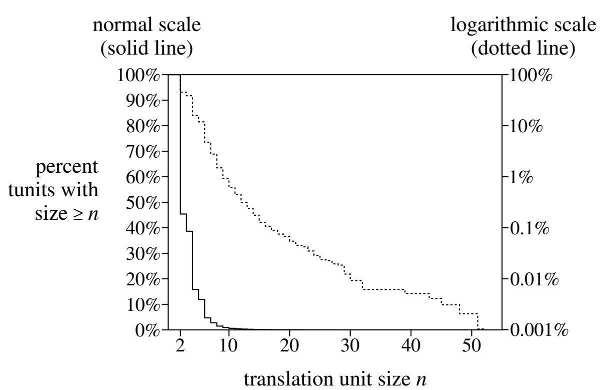

Figure 9: The percentage of translation units in the Copenhagen Danish-English Dependency Treebank with size at least n, plotted on normal and logarithmic scales.

in the parallel treebank, where the size of a translation unit is measured as the number of nodes in the associated complement transfer rule. The extracted transfer rules are useful for many purposes, including machine translation, lexicography, contrastive linguistics, and translation studies, but describing these applications is outside the scope of this paper.

5 Spotting annotation errors

To the human annotator who must check the word-aligned dependency analyses in a parallel dependency treebank, the analyses in Figure 2 and Figure 4 look almost identical. However, from the induced translation units and the associated complement rules shown above, it would have been immediately obvious to the annotator that the analysis in Figure 2 is significantly better than the analysis in Figure 4. This suggests that we can increase the quality of the human annotation in parallel treebank projects by designing annotation tools that continuously compute the induced translation units and present them visibly to the human annotator.

From a linguistic point of view, it can be expected that errors in the dependency annotation will often show up as non-parallelism that results in a large induced translation unit. So in a parallel dependency treebank, we can identify the most egregious examples of non-parallelism errors automatically by computing the induced translation units, and sorting them with respect to their size; the largest translation units will then have a high probability of being the result of annotation errors.

To confirm our linguistic expectation that large translation units are often caused by annotation errors, we have selected a sample of 75 translation units from the Copenhagen Danish-English Dependency Treebank, distributed more or less uniformly with respect to translation unit size in order to ensure that all translation unit sizes are sampled evenly. We have then hand-checked each translation unit carefully in order to determine whether the translation unit contains any annotation errors or not, giving us a data set of the form (C,N) where N is the size of the translation unit and C indicates whether the translation unit is correct ($C=1$) or not ($C=0$). Figure 10 shows our maximum likelihood estimate of the conditional probability $P(C=1|N=n)$ that a translation unit with size n is correct.[6] From the

[6] In order to estimate the conditional probability $p(n) = P(C=1|S=n)$ that a translation unit with size n is correct, we have fitted $p(n)$ to the parametric family $p(n) = \alpha^{n^\beta}$ by means of conditional maximum likelihood estimation with conditional likelihood $L = \prod_{i=1}^{75} p(n_i)^{c_i}(1-p(n_i))^{1-c_i}$. The resulting esti-

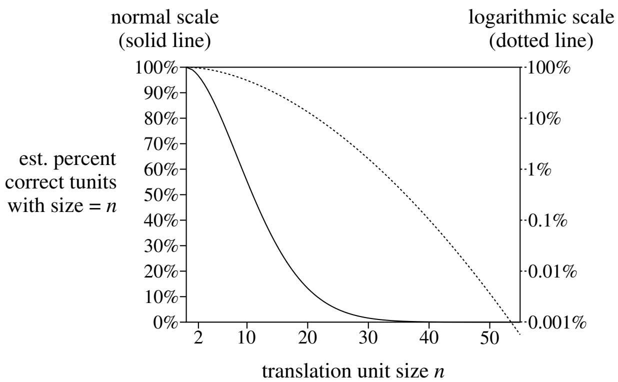

Figure 10: The estimated percentage of translation units with size n that are correct, plotted on normal and logarithmic scales.

graph, we see that the correctness rate decreases quickly with n. For example, only 55% of all translation units with size 10 are correct, and only 13% of all translation units with size 20 are correct. Thus, the statistics confirm that large translation units are often caused by annotation errors in the treebank, so focusing the effort on large translation units can make the postediting more cost-efficient. This also suggests that when developing algorithms for automatic annotation of parallel dependency treebanks, the algorithms can improve their accuracy by penalizing large translation units.

6 Comparing annotation schemes

Translation units can also be used to compare different annotation schemes. This is relevant in parallel treebank projects where there are several possible annotation schemes for one of the languages — eg, because there is more than one treebank or rule-based parser for that language. In this situation, we have the freedom of choosing the annotation schemes for the source and target languages so that they maximize the parallelism between the source and target language annotations. To make an informed choice, we can create a small pilot parallel treebank for each annotation scheme, and compare the treebank annotations qualitatively by looking at their induced translation units, and quantitatively by looking at their average translation unit size. The best choice of annotation schemes is then the combination that leads to the smallest and most sensible translation units.

Since texts are always trivially word-aligned with themselves, the same procedure applies to monolingual corpora where we want to compare two different dependency annotations with each other. In this setup, structural differences between the two monolingual annotation schemes will show up as large translation units. While these structural differences between annotation schemes could have been revealed by careful manual inspection, the automatic computation of translation units speeds up the process of identifying the differences. The method also suggests that the conversion from one annotation scheme to another can be viewed as a machine translation problem — that is, if we can create a machine translation algorithm that learns to translate from one language to another on the basis of a parallel dependency treebank, then this algorithm can also be used to convert from one dependency annotation scheme to another, given a training corpus that has been annotated with both annotation schemes.

7 Conclusion

In this paper, we have addressed the problem that the linguistic annotations in parallel treebanks often fail to correspond to meaningful translation units, because of internal incompatibilities between the dependency analyses and the word alignment. We have defined a meaningful notion of translation units and provided an algorithm for computing these translation units from any parallel dependency treebank. Finally, we have sketched how our notion of translation units can be used to aid the creation of parallel dependency treebanks by using the translation units as a visual aid for the human annotator, by using translation unit sizes to identify likely annotation errors, and by allowing a quantitative and qualitative comparison of different annotation schemes, both for parallel and monolingual treebanks.

mates are $\hat{\alpha} = 0.99$ and $\hat{\beta} = 1.77$ with confidence value 0.87, ie, if a data set D with the same translation unit sizes is generated randomly from the distribution $\hat{p}(n) = \hat{\alpha}^{n^{\hat{\beta}}}$, then the conditional likelihood of D will be larger than the likelihood of our observed data set in 87% of the cases. This means that a two-sided test does not reject that the data are generated from the estimated distribution $\hat{p}(n)$.

8 Acknowledgments

The work was supported by two grants from the Danish Research Council for the Humanities. Thanks to the anonymous reviewers for their helpful comments.

References

Matthias Buch-Kromann, Jürgen Wedekind, and Jakob Elming. 2007. The Copenhagen Danish-English Dependency Treebank. http://www.id.cbs.dk/∼mbk/ddt-en.

Matthias Buch-Kromann. 2006. Discontinuous Grammar. A dependency-based model of human parsing and language learning. Dr.ling.merc. dissertation, Copenhagen Business School. http://www.id.cbs.dk/∼mbk/thesis.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on Multilingual Dependency Parsing. In *Proc. CoNLL-2006*.

A. Cahill, M. Burke, R. O'Donovan, J. van Genabith, and A. Way. 2004. Long-distance dependency resolution in automatically acquired wide-coverage PCFG-based LFG approximations. In *Proc. of ACL-2004*.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2).

Martin Čmejrek, Jan Cuřín, Jiří Havelka, Jan Hajič, and Vladislav Kuboň. 2004. Prague Czech-English Dependency Treebank. Syntactically annotated resources for machine translation. In *Proc. LREC-2004*.

Lea Cyrus. 2006. Building a resource for studying translation shifts. In *Proc. LREC-2006*.

Yuan Ding and Martha Palmer. 2005. Machine translation using Probabilistic Synchronous Dependency Insertion Grammars. In *Proc. ACL-2005*.

Yuan Ding. 2006. *Machine translation using Probabilistic Synchronous Dependency Insertion Grammars*. Ph.D. thesis, Univ. of Pennsylvania.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proc. HLT/NAACL-2004*.

Chung-hye Han, Na-Rae Han, Eon-Suk Ko, and Martha Palmer. 2002. Development and evaluation of a Korean treebank and its application to NLP. In *Proc. LREC-2002*.

Silvia Hansen-Schirra, Stella Neumann, and Mihaela Vela. 2006. Multi-dimensional annotation and alignment in an English-German translation corpus. In *Proc. NLPXML-2006*.

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

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. SPMT: Statistical machine translation with syntactified target language phrases. In *Proc. EMNLP-2006*.

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

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal SMT. In *Proc. ACL-2005*.

Yvonne Samuelsson and Martin Volk. 2006. Phrase alignment in parallel treebanks. In *Proc. TLT-2006*.

K. Uchimoto, Y. Zhang, K. Sudo, M. Murata, S. Sekine, and H. Isahara. 2004. Multilingual aligned parallel treebank corpus reflecting contextual information and its applications. In *Proc. MLR-2004*.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proc. ACL-2001*.