LinES 1.0 Annotation: Format, Contents and Guidelines

Lars Ahrenberg
Department of Computer and Information Science
Linköpings universitet
SE-58183, Sweden
email: lah@ida.liu.se

[ ]

Abstract

This report gives a detailed overview of the annotations used in the LinES Parallel Treebank with examples of their application to actual data. It covers tokenization, morphological, and syntactic analysis. It also covers the principles used for word alignment including a comparison with well-known annotation guidelines such as those of the Blinker project.

1 What is LinES?

LinES is a parallel treebank that has been developed as part of the project *Linguistic micro- and macroanalysis of a translation corpus* with support from The Swedish Research Council (Vetenskapsrådet) in 2004-2005. While this project has now ended, the treebank continues to be developed at a slow but regular pace.

LinES is an abbreviation for *Linköping English-Swedish Parallel Treebank*, the current version of which is 0.9 (March 2007[1]). This version contains data from two sources, Microsoft Access On-line Help and the novel *Jerusalem and back: a personal account* by Saul Bellow. These texts have been collected in earlier projects and constitute a part of the *Linköping Translation Corpus* (Merkel 1999).

The main goal of the LinES project is to develop a resource for the study of translation from English to Swedish, in particular as regards the translation of function words and common syntactic constructions. For this reason, it is important that the syntactic annotation is systematic and accurate. We particularly see LinES as instrumental for the development and evaluation of English-Swedish machine translation systems, as elaborated in (Ahrenberg 2005). An overview of the project and its current status is given in (Ahrenberg 2007).

LinES will be extended with parallel trees from more registers than the two that are currently included, so as to be able to get information on differences in distribution and translation of common constructions. There is no fixed upper limit for the size of LinES, but for the foreseeable future we do not think that it will reach a size that would allow statistical studies of content words and their constructions. For those interested in finding instances of use, LinES will provide a search interface, however, but the instances that are relevant to a particular user will, naturally, be limited in numbers.[2]

[1] The project site is located at http://www.ida.liu.se/∼lah/transmap/Corpus/

[2] The English-Swedish Parallel Corpus, ESPC, developed as a joint project by the universities of Lund and Göteborg, is larger but have restricted access. For information, see http://www.englund.lu.se/content/view/66/127/.

This report describes the annotation system used in LinES and some of the general guidelines and considerations for its application to actual data. It covers tokenization, morpho-syntactic analysis and word alignment. It should be noted that the guidelines have changed over time and that the annotation of LinES 0.9 does not always respect them. Thus, for version 1.0 of the treebank itself, the current annotation is being revised.

1.1 Background to the annotation

There are some general considerations that have played a role for the design of LinES annotation. First of all, LinES is a bilingual resource, so we have made an effort to use the same categories and features for both English and Swedish, and as far as possible the same principles for tokenization and annotation. Thus, segments are tokenized so as to make tokens correspond one-to-one, unless there are overriding arguments against such a choice. There is one set of part-of-speech categories used for both languages, and they are applied to tokens so that corresponding tokens are categorized the same way (again, unless there are strong counter-arguments). Similarly, the structural syntactic analysis is performed using a joint set of dependency relations. However, since the morphology of the two languages differ, there are morphological features that apply only to one of the languages, but when the same feature is used for both languages, it is applied uniformly.

Second, to speed up annotation we have used the Functional Dependency Grammar (FDG) parsers from Connexor Oy for both English and Swedish.[3] Thus, we have wished to make use of the linguistic information that these parsers provide as much as possible. In particular, this means that the syntactic annotation is based on dependency relations, not phrase structure. However, this consideration is not always congruent with the one just mentioned, as the parsers do not use the same categories and relations, and do not always treat corresponding words, that could be regarded as having the same part-of-speech, in the same way. In these cases, the wish for parallellism in annotation takes precedence, and categories have been harmonized.

Third, our annotation is largely structurally oriented. This has effects both for the monolingual annotation and for alignment. For example, where the FDG parsers use dependency relations that suggest semantic categorisations such as `loc` or `tmp`, LinES annotation is satisfied with relations such as adverbial (`advl`) and modifier (`mod`). Where there is a choice between a structure-oriented or semantics-oriented characterization of a token, LinES usually prefer the first. Thus, subjects of small clauses such as *him* in *We let him go.* are treated as objects of the matrix verb and relative pronouns such as *that* in a phrase such as *the car that passed by* are annotated as a subordinate-clause marker (`subm`) rather than as a subject. In alignment, we regard it as a sufficient condition for two tokens to be aligned, if they fill corresponding structural positions. For instance, a *when* may be aligned to Swedish *om* (if), if they introduce corresponding clauses.

1.2 Structure and formatting

A subsection of LinES consists of three files: a source file, a target file, and a link file.

Source and target files of LinES are monolingual files that are formatted in XML according to the document type definition `liu-mono.dtd`. All elements and attributes allowed by liu-mono.dtd are not used, however, primarily for the reason that the files are viewed as sentence

[3] The parsers are now called Machinese Syntax. For more information, see `http://www.connexor.com/software/syntax/`.

```
<s id=”s412”>
<w id=”w6894” relpos=”1” base=”of-course” func=”advl” fa=”4” pos=”ADV”>Of-course</w>
<w id=”w6895” relpos=”2” base=”it” func=”subj” fa=”3” pos=”PRON” msd=”NOM-SG”>it</w>
<w id=”w6896” relpos=”3” base=”must” func=”v-ch” fa=”4” pos=”V” msd=”PRES”>must</w>
<w id=”w6897” relpos=”4” base=”do” func=”main” fa=”0” pos=”V” msd=”INF”>do</w>
<w id=”w6898” relpos=”5” base=”financial” func=”attr” fa=”6” pos=”A” msd=”ABS”>financial</w>
<w id=”w6899” relpos=”6” base=”news” func=”obj” fa=”4” pos=”N” msd=”NOM-PL”>news</w>
<w id=”w6900” relpos=”7” base=”and” func=”cc” fa=”6” pos=”CC”>and</w>
<w id=”w6901” relpos=”8” base=”sports” func=”cc” fa=”6” pos=”N” msd=”NOM-PL”>sports</w>
<w id=”w6902” relpos=”9” base=”well” func=”advl” fa=”4” pos=”ADV”>well</w>
<w id=”w6903” relpos=”10” base=”enough” func=”mod” fa=”9” pos=”ADV”>enough</w>
<w id=”w6904” relpos=”11” base=”.” pos=”FE” msd=”Period”>.</w>
</s>
```

Figure 1: A sentence analysis in LinES monolingual XML format.

collections rather than as texts. The most important features of this used format are the following:

- A monolingual file is structured in terms of segments and tokens. Segments are demarcated by <s>-tags and tokens by <w>-tags.

- A segment need not be a sentence in the grammatical sense; it may be smaller, such as a noun phrase, or larger, such as a sequence of two or more sentences. Segments are determined by the punctuation of the original texts and the segment alignment. Thus, every segment of a source file corresponds to a segment in the target file.

- Each segment has a unique identifier, its s-id. The first segment of a file is assigned the identifier s1, and the following segments are assigned identifiers s2, s3, and so on. Corresponding source and target segments are assigned identical segment identifiers, though they occur in different files.

- Each token has a unique identifier, its token-id. Token identifiers consist of a string that starts with a w followed by a number. The numbers are consecutive in the file, so that the first token of the first segment has token identifier w1 and the following tokens have identifiers w2, w3, and so on.

- Tokens carry a number of attributes for annotation. These are explained in detail in the following sections.

Figure 1 illustrates the format and contents of XML markup in LinES.

2 Guidelines for tokenization and lemmatization

To a large extent tokenization and lemmatization is done in the same way as in the parsers. However, mistakes have been corrected, when found, and there is no annotation of the internal structure of Swedish compounds, which is something that the Swedish parser attempts to identify as part of the lemmas. In the monolingual files, tokens are demarcated by <w>-tags while lemmas are given as values of the attribute `base`.

2.1 Tokenization

Usually, an orthographic word corresponds to a token, but in some cases a single word may introduce two tokens. For English this happens in the following cases:

- A pronoun or noun followed by a clitic verb form such as *he's*, *I'm*, or *we're* is split into two parts, one for the pronoun and one for the verb form.
- An auxiliary verb with a cliticized negation is separated into two tokens. For example, *won't* is tokenized as *wo* (base-value: will), and *n't*, *cannot* as *can* and *not*.
- All punctuation marks are represented as separate tokens. For example, a word occurring with a comma is split into the word and the comma.

Note that nouns with a genitive suffix ('s or ') are not split into two tokens. The same principles apply to Swedish, but clitics are much rarer in the Swedish texts, so it is mainly the third principle that takes effect.

There are also instances of two or more orthographic words being rendered as single tokens. We would like to keep the number of such instances low, as it is hard to achieve consistency and good performance at the same time for an automatic tokenizer. While there are still other instances of such multi-word tokens in the data, the following list enumerates the cases for which we think a multi-word analysis is justified.

- The Swedish compound determiners and pronouns *den här*, *den där*, *de här*, *de där*, *en del* corresponding to English *this*, *that*, *these*, *those* and *some*.
- The Swedish emphasizing word group *som helst*,
- A number of Swedish compound adverbs such as *i dag, i går, i morgon, ut och in, fram och tillbaka, över huvud taget*; many of these are, in fact, sometimes written as one orthographic token,
- A number of Swedish compound prepositions such as *på grund av, i stället för, med hjälp av*, ...
- The English compound determiners such as *a few, a bit.*
- The English compound pronouns such as *one another, each other, something else, anything else.*
- A number of English compound prepositions such as *because of, instead of, according to, in order to*, ...
- A number of English compound adverbs such as *of course, at once, in and out*, *back and forth*, *now and then, side by side*, ...

2.2 Lemmatization

The following principles guide the determination of lemmas:

- The lemma for nouns is represented by the singular, unmarked form. In Swedish, which has both definite and indefinite forms, the indefinite form is used.
- The lemma for verbs is represented by the infinitive, active form.

- The lemma for adjectives is represented by the absolute, or positive form.
- The lemma for determiners is represented by the singular, non-neuter form,
- The lemma for a personal pronoun is represented by the nominative form. This means that accusative forms such as *me* or *him*, and genitive forms such as *my* or *their* are assigned the nominative form as their lemma.
- The lemma for a multi-word token is represented by a hyphenated compound of the different parts.

3 Parts of speech

The parts-of-speech used in LinES includes the common lexical categories with some extra categories for special text tokens. Altogether LinES 1.0 uses 22 categories, which are listed in Table 1.

As pointed out in the introduction, LinES uses a common set of parts of speech for both languages. Some distinctions are made in LinES, which are not made by the parsers, for example distinguishing between proper and common nouns. On the other hand, LinES employs Participle as a common part-of-speech, where the parsers treat different sub-types of participles as their own parts-of-speech. In addition, many individual words have been given different parts-of-speech in LinES. For example, words that introduce subordinate clauses, such as English *when*, Swedish *när* are classified as subjunctions in LinES rather than as adverbs or pronouns.

As illustrated in Figure 1, the part-of-speech for a token is registered under the attribute `pos` in the monolingual files.

The assignment of a part-of-speech to a token is not always self-evident. The following sections discuss some of the decisions we have made.

3.1 Nouns

Proper nouns (`PN`) are generally distinguished from common nouns in both English and Swedish by being capitalized and by not carrying a definite article when used for definite reference. The category is primarily used for single tokens that constitute names in themselves. It is also used for the individual parts of multi-word names, where the status of the multi-word unit as a proper noun is well established. For instance, *New York* is analysed as a sequence of two proper nouns. However, complex names that retain their shape as descriptions are not analysed as a sequence of proper names. For example, while the abbreviation *FBI* is analysed as a proper noun, *The Federal Bureau of Investigation* is analysed as a sequence of a determiner, an adjective, a noun, a preposition, and a noun.

3.2 Abbreviations

Abbreviation `ABBR` is a category used quite sparsely. If the abbreviation is used as a name or a nominal, the category `PN` is preferred. If it is abbreviating an adverb, the category `ADV` is preferred. The only case where `ABBR` is used, is when an abbreviation is defined, explicitly or by association with the full description.

3.3 Adjectives, determiners, and numerals

Adjective (`A`) is used for words that can serve both an attributive function in a noun phrase, and as a predicative. The English forms ending in *-ly* and the Swedish forms ending in *-t*

Category symbol	**Description**	**English examples**	**Swedish examples**
A	Adjective	*tall, taller*	*lång, längre*
ABBR	Abbreviation		
ADV	Adverb	*up, not, where*	*upp, inte, var*
CC	Conjunction	*and, or*	*och, eller*
CCI	Initial Conjunction	*both, either*	*både, antingen*
CD	Code symbol	*WHERE*	*WHERE*
CS	Subjunction	*while, that*	*medan, att*
DET	Determiner	*the, which*	*den, vilken*
FE	Ending punctuation	*., ?*	*., ?*
FI	Internal punctuation	*, ;*	*, ;*
FP	Paired punctuation	*(,)*	*(,)*
IJ	Interjection	*ouch, no*	*aj, nej*
INFM	Infinitive marker	*to*	*att*
N	Noun	*thing*	*sak*
NUM	Numeral	*five, third*	*fem, tredje*
PCP	Participle	*working, worked*	*arbetande, arbetad*
PN	Proper noun	*London*	*London*
POSP	Postposition	*ago*	*runt*
PREP	Preposition	*in, behind*	*i, bakom*
PRON	Pronoun	*you, something*	*du, någonting*
SYM	Symbol	*%, :)*	*%, :)*
V	Verb	*works, went*	*arbetar, gick*

Table 1: The parts-of-speech used in LinES.

English word	Swedish counterparts	Coding
another	-	DET:IND
-	annan	DET:ADJ
other	andra, annat	DET:ADJ
same	samma	DET:ADJ
last	sista, förra	DET:ADJ
next	nästa, näste	DET:ADJ
own	egen, eget, egna	DET:ADJ
such	sådan, sådana	DET:ADJ
many	många	DET:IND
more	fler, mer	DET:IND
most	flest, flesta, mest, mesta	DET:IND
much	mycket	DET:IND
few	få	DET:IND
fewer	färre	DET:IND
several	åtskilliga	DET:IND

Table 2: Words classified as adjectival or indefinite determiners.

that are derived from adjectives but serve adverbial functions are classified as adverbs (`ADV`). Many comparative and superlative forms of adjectives can also stand in adverbial relations to verbs; these forms are then also classified as adverbs.

Ordinal numbers, such as *first, third*, that also may be either attributive or predicative, are classified as numerals (`NUM`) and subcategorized as ordinals (`ORD`). Other ordinal words such as *next, last*, on the other hand, that do not have a corresponding cardinal number are classified as determiners (`DET`).

There are many words that impose a definite or indefinite interpretation on a noun phrase that they are part of, and thus are on the borderline between adjectives and determiners. We have mostly opted for classifying them as determiners, but use a subcategorization feature `ADJ` to mark their similarity with adjectives. If the meaning is indefinite, the subcategorization feature `IND` is used. A list of such words is given in Table 2.

3.4 Pronouns

Tokens are only classified as pronouns (`PRON`) when they contract a function that is typical of a head noun such as subject, object or prepositional complement. There are many such tokens that can also modify a noun. When doing so, they are classified as determiners instead. Thus, there are many tokens that, out of context, are ambiguous. We allow this ambiguity since there are tokens that are unambiguously determiners, such as the articles, and other tokens that are unambiguously pronouns, such as the masculine and feminine personal pronouns. Also, in English we have a formal distinction between the genitive, independent pronouns *mine, yours, ...* and the dependent ones *my, your, ...* that motivates a distinction between the two categories.

Words that can be classified as either determiner or pronoun usually have the same set of subcategorization features. We distinguish the following subtypes of pronouns: personal, demonstrative, adjectival, indefinite, existential, reflexive, reciprocal, relative, and wh.

Pronominal adverbs such as *where* and Sw. *där, varför* are classified as adverbs rather than as pronouns.

3.5 Coordinating conjunctions

Typically, a coordinating conjunction is infixed between two items that it relates. These tokens are classified as `CC`. However, some coordinating conjunctions come in pairs, such as *both-and* or the equivalent Swedish *både-och.* The first of these conjunctions are then categorized as `CCI`. When a coordinating conjunction has become part of a fixed expression that ends a list of conjuncts, such as English *and so on*, Swedish *och så vidare*, often abbreviated *osv.*, they are categorized as adverbs.

When a coordinating conjunction introduces a new segment, it usually refers back to a previous segment, which is not part of the current pair of translations. It is still categorized as `CC` and treated as belonging to the segment in which it occurs.

3.6 Subjunctions and prepositions

The basic rule is that a subjunction introduces a subordinate clause, while a preposition introduces a noun phrase. Some of the prepositions may also introduce adjectival or adverbial phrases, and even infinitival or participial verb phrases, e.g., *with* and *without* and, if so, they are still categorized as prepositions. However, there are also words that may introduce clauses as well as phrases, such as the comparative subjunctions *than, as, like* and their Swedish counterparts *som, än.* We regard such words as prepositions when they introduce a phrase, and as a subjunction when they introduce a clause.

3.7 Participles

Participle (`PCP`) is used for words that morphologically are clearly derived from verbs and that have a verbal function. We subcategorize them by their normalised endings, in English ING and EN, in Swedish NDE and AD. The EN-participles in English are further categorized according to their function as a passive or a perfect use. Thus, what the Connexor parsers regard as parts of speech, we regard as sub-categorizations. A complication for Swedish is that the so-called perfect (AD) participles are inflected for number and gender in very much the same way as adjectives, and that many adjectives have the same endings. Still, we maintain a difference between participles and adjectives in these cases, and use the criteria on form and function as a basis. Another criterion, in more involved cases, is that the dynamism of a verb is kept by a participle, but not present with an adjective. Thus, the word *komplicerat* in the sentence *Problemet är komplicerat* (Eng. The problem is complicated) is analysed as an adjective, in spite of the fact that a related verb *komplicera* (complicate) exists. However, the normal interpretation is that the word denotes an inherent property of the problem, as it were, not one inflicted upon it by an agent of some sort. In this it can be contrasted with the sentence *Problemet är löst* (The problem is solved) where an agent must be assumed and the word *löst* is thus analysed as a participle.

4 Morphological subcategorization

The attribute `msd` (for morpho-syntactic description) in a monolingual file registers information pertaining to subcategorisation of tokens. The value of the `msd`-attrubute may denote a complex of properties. For instance, the noun *news* is annotated as follows:

<w ... base=”news” pos=”N” msd=”NOM-PL” ... >news</w>

The value of the msd-attribute is a concatenation of the two values NOM (for nominative case), and PL (for plural number), with a hyphen (-) as the concatenation marker. Note that the property dimensions are left implicit in the annotation. Note also that no value is required on the msd-attribute, even if the part of speech is one that is inflected or subcategorized in other ways.

Table 3 gives an overview of all the individual feature values that can appear on a `msd`-attribute, together with the parts-of-speech with which they can combine.

5 Dependency functions

As shown in Figure 1 the dependency relations are coded as functions from tokens to tokens, using the segment internal index, i.e., the value of the attribute `relpos` to represent the token. The head, or governor, of a token is coded in the attribute `fa`, while the type of dependency is coded under the attribute `func`. Thus, the information func=”attr” fa=”6” associated with fifth token *financial* in Figure 1, means that this token has been assigned the sixth token *news* as its governor, and given the role of attribute in relation to it.

The value ’0’ is used as a value for the `fa`-attribute, when a token is analysed as the main governor of the segment. Punctuation tokens are not assigned dependency relations. All of this follows the representation format used by the Connexor parsers.

An overview of the function labels used is given in Table 4.

The dependency relations can be divided into three categories that reflect their application. The *general relations* are not restricted to any type of syntactic unit, but have both phrases and clauses in their range. In this category we count `main, cc, app` and `initm`. The *clause level relations* commonly relates the head word of a phrase to a verb heading a clause. To this category belongs `subm, subj, obj, pobj, sc, oc, advl, prt, top, ext` and `vch`. Finally, the *phrase level relations* relate to phrasal heads and comprise the following relations: `attr, ad, det, mod, amod, pcomp`.

The three functions `amod, ext, top` form a set of their own. They are used to register non-projective relations, i.e., situations where the proper head of a dependent is inaccessible because of the requirement for projective dependency structures. These relations make the representations pseudo-projective in the sense of (Kahane et al 1998).

In the following sections, the use of the most important relations are explained.

5.1 General functions

5.1.1 The segment governor (`main`)

Every monolingual segment is required to have exactly one topmost governor. This topmost token is assigned the function `main`.

If a segment consists of one ore more clauses the topmost token will generally be a verb. In case the segment consists of several coordinated clauses, the first one will be taken to include the main verb. When a clause contains several verbs that form a verb chain, we follow the rule that the first one that subcategorizes the subject should be taken as the main verb. In practice, this means that a finite set of verbs including *be*, *have* and modal auxiliaries are not main verbs when followed by another verb. Thus, these verbs are assigned the main function only in case no other verb with more semantic content follows.

In direct speech, the speech act verb is assigned the main function, whereas the utterance is assigned the `obj`-function in relation to that verb.

Value	**Description**	**Range**	**Languages**
ABS	Absolute	ADJ, ADV	EN and SE
ACC	Accusative	PRON	EN and SE
ACT	Active	V	EN and SE
AD	Passive participial	PCP	SE
ADJ	Adjectival	DET, PRON	EN and SE
AUX	Auxiliary	V	EN and SE
CARD	Cardinal	NUM	EN and SE
CMP	Comparative	ADJ, ADV	EN and SE
DEF	Definite	DET, N	DET: Both; N: SE
DEM	Demonstrative	DET, PRON	EN and SE
DEP	Dependent	PRON	EN
ENG	English word	Any	SE
EX	Existential	PRON	EN
FGN	Foreign word	Any	EN and SE
GEN	Genitive	ADJ, N, NUM, PN, PRON	EN and SE
ID	Identity	NUM	EN and SE
IDP	Independent	PRON	EN
IMP	Imperative	V	EN and SE
IND	Indefinite	DET, PRON, N	Both, except N: SE
INF	Infinitive	V	EN and SE
ING	Gerundive	PCP	EN only
NDE	Present participial	PCP	SE only
NOM	Nominative	ADJ, N, NUM, PN, PRON	EN and SE
ORD	Ordinal	NUM	EN and SE
PASS	Passive	V, PCP	PCP: EN; V: SE
PAST	Past tense	V	EN and SE
PL	Plural	ADJ, DET, N, PN, PRON	Both, except ADJ: SE
PERF	Perfect	PCP	EN
PERS	Personal	PRON	EN and SE
PRES	Present tense	V	EN and SE
RCP	Reciprocal	PRON	EN and SE
RFL	Reflexive	PRON	EN and SE
SBJ	Subjunctive	V	EN and SE
SG	Singular	ADJ, DET, N, PN, PRON	Both, except ADJ: SE
SPV	Superlative	ADJ, ADV	EN and SE
SUP	Supine	V	SE only
WH	Wh-form	ADJ, ADV, DET, PRON	EN and SE

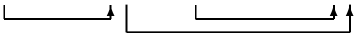
Table 3: Feature values registered under the attribute `msd` and used for subcategorization in LinES.

Function	**Description**	**Domain**	**Range**
ad	Phrase level Adverbial	ADV	ADJ, NUM, N
advl	Clause level Adverbial	ADV, PREP	V, PCP
amod	Attribute modifier	PREP, V	N, PN
app	Apposition	ANY	N, V
attr	Attribute	ADJ, NUM, N	N, PN
cc	Coordinator	CC	ANY
det	Determiner	DET	N
ext	Extraposition	ANY	V
initm	Utterance initiator	IJ	ANY
main	Main	ANY	–
mod	Modifier	ADV, PREP, V, PCP	ADJ, N, PN
obj	Object	N, PN, PRON	V, PCP
oc	Object Complement	ADJ	N, PN, PRON
pcomp	Prepositional Complement	N, PN, PRON	PREP
pobj	Prepositional Object	PREP	V, PCP
prt	Particle	ADV, PREP	V, PCP
sc	Subject Complement	ADJ, N	V
subj	Subject	N, PN, PRON	V
subm	Subordinator	CS	V
top	Topic	N, PN	V
vch	Verbal Chain Item	V	V, PCP
voc	Vocative	PN, PRON, N	V

Table 4: Dependency functions used in LinES. Note that the values given for domain and range should be taken as typical rather than complete.

If a segment consists of a phrase, the `main` function will be assigned the token that can be considered the governor of the phrase. In the case of prepositional phrases, this token will be the preposition. In a noun phrase it will usually be a noun, and when there is a sequence of nouns, including proper nouns and pronouns, we generally choose the last noun in the sequence as the governor. Thus, in a phrase such as (1) the token *Jobert* is regarded as the head.

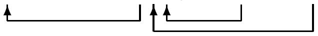
(1) *foreign minister Maurice Jobert*



5.1.2 Coordination (`cc`)

The function `cc` is used for both coordinating conjunctions, and the conjuncts they introduce. The main rule, not always implemented in the parsers, is that a conjunction as well as the head word of the phrase it introduces, is attached to the nearest head word to the left that it is coordinated with. Thus, in our analysis of (2) the first token *Denmark* is assigned the subject function with respect to the main verb *are*, while *Norway* contracts a `cc`-relation to *Denmark*, and *and* as well as *Sweden* contract `cc`-relations to *Norway.*

(2) *Denmark, Norway and Sweden are very similar countries*



There is one exception to this rule, and that is when the first conjunct is the first item of a compound. Thus, in a phrase such as Sw. *hund- och kattmat* (dog food and cat food), the first word is treated as a conjunct of the last, and the conjunction attaches to the right.

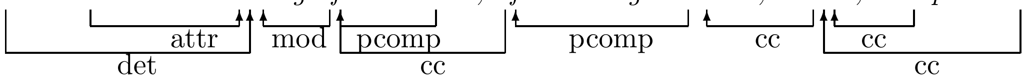
Several conjunctions, such as *and* or *but*, can appear at the beginning of a segment without any token to attach them to, since that token is not part of the corpus, or, if it is, belongs to a different segment. When this happens they are assigned the function `ad` and the head is taken to be the head of the phrase or clause they introduce.

It is not uncommon for coordinations to include a large number of conjoined phrases, as in the following example:

(17) *is there in this world, by now, a natural understanding of revolution, of mass organization, cadres, and police rule?*

While there is an explicit conjunction, *and*, with the last conjunct, none of the other conjuncts are so marked. Since we regard prepositions as heads of prepositional phrases we analyze this coordination as having two levels. Thus, the second *of* is attached to the *of* which is head of the first conjunct *of revolution*, while the rest, from the word *organization* on, constitute a coordination of nouns. The analysis is depicted below:

(18) *a natural understanding of revolution, of mass organization, cadres, and police rule.*

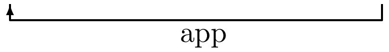


5.1.3 Apposition (`app`)

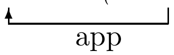
The function `app` is used for head tokens that govern a phrase that either (i) is inserted as extra material within a syntactically coherent word sequence, or (ii) added as a kind of afterthought. A common sign of an apposition is the use of parentheses to mark an insertion. If, on the other hand, there is a syntactic link such as a preceding conjunction or subjunction that initiates the phrase, the function is not used. Instead the phrase should be analysed as a conjunct (`cc`) or a modifier (`mod`).

In the following examples, the tokens marked in bold face have been analysed as appositional tokens:

(3) *But I succeed in supressing this – a* **triumph** *over myself.*



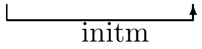
(4) *We cannot avoid being politicized (to* **use** *a word as murky as the condition it describes) ...*



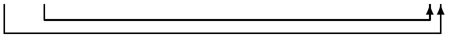
5.1.4 Initiating marker (`initm`)

In dialogue, utterances often start with a short word or phrase that either gives feedback to the previous speaker, or else is a sign of what is sometimes called own communication management. The relation of this word or phrase to what follows is loose, syntactically, and we thus employ a special function for it, `initm`, which may be read as *initial marker* or (utterance) *initial management token.* The following are some examples.

(5) **No***, I do n't think so.*



(6) **Well, yes** *well-informed people do have this information in their files.*



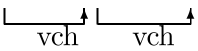
As illustrated in (6), several tokens in sequence may be given the `initm`-function.

5.2 Clause-level functions

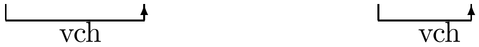
5.2.1 Verbal chain items (`vch`)

Verbal chain items (`vch`) are the modal auxiliaries, the infinite markers *to* and *att*, respectively, and forms of the verbs *be* and *have* and their Swedish counterparts, when they have another verb or chain element as a governor. Verbal chains may consist of several chain items and, as a rule, a chain element depends on the first verb or chain item to its right. In both English and Swedish the verbal chain may be broken up by a light adverb such as the negation. The following are examples of verbal chains; (7) is an unbroken chain, while (8) is broken up by the negation.

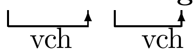
(7) *They* **have been watching** *us for a long time.*



(8) **Did** *n't he* **tell** *her that he* **was coming** *tonight?*



(9) Sw. *Du* **kommer att ångra** *dig.*



5.2.2 Particles (`prt`) and complex verbs

Both Swedish and English have complex verbs of the type *give up; ge upp* where the second item is usually a light adverb or preposition. The construction seems to be more common in Swedish, where also nouns or adjectives may constitute the second part, and even three-part constructions such as *ge med sig* (lit. give with oneself) are quite frequent. Moreover, reflexives may occur without a particle as in *ge sig* 'give in'.

If the light second part carries a stress and/or is followed by a bare object, it is analysed as a particle and marked as `prt`. On the other hand, if the interpretation of the compound is compositional with a literal interpretation of the adverb as in *stay behind, come home* the second part is analysed as adverbial. The reflexive is analysed as object (`obj`) when it occurs on its own, and as a complement of the preposition (`pcomp`) if it follows a preposition.

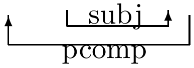
5.2.3 Subject (`subj`)

The subject function (`subj`) is used both for formal subjects and ordinary subjects. Thus, we allow two tokens to carry the `subj`-function in clauses such as *There is a man in the garden.* Both of them are assigned the same governor, i.e., the finite verb.

It should be noted that the governor of a subject token is the finite verb of a clause rather than the main verb, when there is a difference. This is the same convention as used by the parsers and is motivated by the assumption that the complete verbal chain can be regarded as a complex predicate.

If there is no finite verb, but a participial construction to which a noun phrase may be seen as providing a subject, the `subj`-function can still be applied. Also, if a verb has been gapped but their is a predicative construction of some sort, we use `subj` and let it take the predicative as its governor. An example is the following:

(10) *With* **John** *absent, the atmosphere was quite different.*



The English relative pronouns *that, which* and *who*, the Swedish relative pronoun *som* are not assigned the subject function, since we take them to be indifferent to argument roles. This is different from the way they are handled by the parsers. Instead they are taken to be clause initiators and assigned the `subm`-function.

5.2.4 Subject complement (`sc`)

Subject complements are predicatives that normally follow a copulative verb such as *be* or *grow* and apply to the subject. As shown in example (10), if the copula is absent, the predicative will be counted as the governor of the subject and will itself assume a relation that depends on its context.

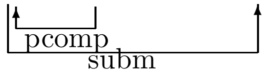
5.2.5 Subordinate clause marker (subm)

The subm-function is primarily assigned subjunctions that introduce a subordinate clause, including adverbial, nominal or relative clauses.

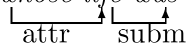
A signalled relative clause is marked in its beginning by a subjunction, or a relative pronoun or adverb. Whatever the case, the function subm will be used, rather than an argument function such as subj or obj. As for part-of-speech we treat Eng. *that* and Sw. *som* as subjunctions (CS), while relative wh-words are treated as pronouns, determiners, or adverbs and marked in the morphological analysis as both WH and REL. It is the main verb of the clause that is considered to be its head and assigned the mod-function with respect to the noun that is modified.

When there is a complex relative phrase, its head is assigned the subm-function, while the analysis of the relative word will depend on the head. The following illustrate the common cases of a preposition or an attribute as the first token of the relative clause.

(11) *a place in which she could live*



(12) *a man whose life was very easy*



5.2.6 Non-signalled coordination or subjunction

When two units are juxtaposed without an explicit coordination marker, there are several alternative analyses possible. If a coordinating conjunction follows, as illustated above, the cc-analysis is appropriate, but if there is only a punctuation marker such as a semicolon or a full stop, the choice can be difficult. We use the following general guidelines: If the relation between two clauses can be made explicit by a subjunction or adverb that indicates an adverbial relation, then annotate the relation as advl. Should the most proper linking word be *and* or *but*, however, the analysis should be cc. Finally, if the relation is that of an afterthought, or a comment pertaining to some constituent, it should be analysed as an apposition (app).

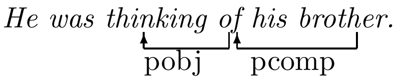
This type of problem arises in particular, when there is a 1-2, or 2-1, relation at the sentence level. This is so, because we only allow one token to carry the main attribute in any segment, even if it is made up of two sentences.

A special case are the Swedish conjunctions *för* and *ty* meaning 'because' or 'since'. The clause following either of these has main clause word order, why they are taken to be coordinating conjunctions. However, since the meaning they express is causal we analyse that clause as having an adverbial relation to the matrix clause, and the conjunctions themselves as carrying the dependency function subm.

5.2.7 Complements (obj, pobj, oc)

Noun phrases and heads of clauses and verb phrases that are related to a verb or an adjective as complements are assigned the obj-function. Complements that are preceded or introduced by a preposition are assigned the pobj-function. Note that it is the preposition that is assigned the function pobj, while the head noun of the noun phrase is regarded as a prepositional complement, (pcomp). See example (13) for an illustration.

(13) *He was thinking of his brother.*



The oc-function, for object complement, is assigned to a predicative that applies to an object. This happens only with a few verbs such as *make* and *consider*.

Note that there is no special function for indirect objects. Thus, with bitransitive verbs such as *give* there will be two tokens carrying the obj-function. A benefactive complement introduced by a preposition will be regarded as a prepositional object.

5.2.8 Adverbial (advl)

The advl-function is used for all kinds of clause-level adjuncts, i.e., single words and phrase heads that are not part of the verbal chain, nor is an argument of the main verb. Thus, this function covers adjuncts with a broad range of meanings such as temporal, spatial, causal, conditional, quantitative and manner.

Adverbials may be headed by adverbs, prepositions, nouns or verbs. Thus, the head of an adverbial clause is considered to be the main verb of that clause, not the subjunction.

5.2.9 The non-projective functions top and ext

Some relations cannot be captured within a projective framework. This applies to many cases of what in transformational parlance is called movement. In LinES, the function top ("topicalized") is used for units that have been moved to the front of a clause, in particular a main clause, while ext ("extraposed") is used for units that have been moved to the end. These labels do not, of course, give representation to the real governor of the units that have been moved, but they signal that there is a non-normal relation with the governor that it has been assigned to.

(14) Him I remember	*Him* is obj wrt *remember*
(15) Him I think I remember	*Him* is top wrt *think*
(16) I met a man that I've seen before	*seen* is mod wrt *man*
(17) I met a man yesterday that I've seen before	*seen* is ext wrt *met*

Note that movement may also occur without breaking projectivity. If so, normal rules apply, and some of the ordinary clause-level functions are used. Examples (14)-(17) illustrate the use of these functions.

5.3 Phrase level functions

Phrase level functions relate a phrasal constituent to the phrase head. The function pcomp is special: its head is always a preposition and it applies to the token which is the head of the governed phrase, usually a noun phrase. The functions det and attr apply only to constituents of noun phrases, while the functions ad and mod can be used with constitutents of phrases of any type.

5.3.1 Determiners and attributes

The primary difference between the functions det and attr is that det is used with determiners, i.e., tokens having the part-of-speech DET while attr is used with all other tokens that precede and modify the head noun of a noun phrase. In a way, this makes the function `det` superfluous, but it is used nevertheless.

The `attr`-function can apply to adjectives, common and proper nouns, and pronouns. Note, in particular that possessive pronouns such as *my* and *his* are assigned the function `attr` in phrases such as *my wife*, *his job.* The same applies, naturally, to full genitive noun phrases modifying a head noun.

5.3.2 Phrase-level adverbial (`ad`)

The `ad`-function primarily applies to phrase-levels adverbs that precede and modify a phrasal head. The part-of-speech of the phrasal head may vary. Thus, we find adverbs that have an adjective as head (*completely blank, very bright, just perfect*), another adverb (*too much, very likely*), a numeral (*only fifteen, approximately fifty*), a preposition (*back in town, down to the beach*), a noun or pronoun (*not him, just an accident*).

Heads of phrases can also be assigned the `ad`-function, when they precede and modify a governor. Examples are *five years old, two meters tall* and the preposition in Swedish modifiers of the type *av staten bortglömda, med balkar försedd* ('by the state forgotten', 'with beams furnished').

A conjunction that appears in the beginning of a segment is assigned the `ad`-function. Its head will be the head of whatever comes after. If it is a clause, the head will be the main verb of the clause.

5.3.3 Modifiers (`mod, amod`)

The function `mod` is used for modifiers of a phrase that appear after the head. Thus, prepositional attributes and relative clauses, including participial and infinitive constructions, are assigned the `mod`-function, as are adverbs when they follow the head word, as in *good enough.*

However, if the modifier is related to an attribute preceding the head, the function `amod` ("attribute modifier") should be used. A typical case is given by comparative modifiers as in *She is a better player than you* where the proper head of the *than*-phrase is arguably *better* rather than *player.* Still, since the noun is the governor of the adjectival attribute we have to assign the noun as the head also of the post-modifier and use `amod` to indicate that the proper governor is an attribute.

Arguments of adjectives as in *afraid of rats* are not considered modifiers, but prepositional objects, while those rare cases of arguments that are not headed by a preposition, as in *worth a million* are regarded as objects, `obj`.

Modifiers should also be separated from post-positions, which are heads of their phrase. For instance, we treat *ago* as in *three years ago* as a post-position rather than as a modifying adverb. Similarly, in Swedish *jorden runt*, 'around the earth', *runt* is analysed as a post-position and *jorden* as a `pcomp`.

6 Word alignment

The purpose of the word alignment is to provide a basis for the description and characterization of the translations in the corpus. Since the syntactic annotation employs dependency relations, any registered word alignment between two tokens – or, as we will say for short: any link – provides a basis for several types of relations:

- A lexical correspondence of the source language token with the target language token,
- A part-of-speech correspondence,
- A correspondence of morphological features,
- A correspondence of dependency functions,
- Structural correspondences of the sub-tree headed by the source language token with the sub-tree headed by the target language token.

Now, we cannot always take the last type of correspondence for granted, at least not if we wish it to be perfect, since some dependent of the the source language token may not have a correspondent dependent on the target language token; it may be untranslated or translated by something outside of the sub-tree identified by the target language token. Nevertheless, that structural correspondence is of potential interest, even if it is not perfect.

In fact, there are many structural correspondences that may be of interest. The first one is a single dependency relation and its image on the other half of the treebank. Given that such correspondences can be found, correspondences for all other structures can also be found.

6.1 Basic principles

Since structural correspondences can be obtained from the dependency trees, we regard the token correspondences as primary and generally wants them to be as small as possible in terms of the number of units involved. Ideally, they should be 1-1, 1-0, or 0-1. This is not always possible, however, but if there are no stronger criteria that apply, we prefer small links. Note, in particular, that null links are included when sub-trees are sought for. For instance, the very common case of an English definite noun phrase such as *the house* being translated with a single definite Swedish noun such as *huset* will be aligned with a null (1-0) link for the determiner and a 1-1 link for the nouns. Thus, at the token level we can retrieve the lexical correspondence *house* ∼ *hus*, at the structural level we can retrieve correspondences, such as `[the N]` ∼ `[N]`.

All alignments in LinES respect token boundaries. With a few exceptions, such as clitics, token boundaries are orthographic boundaries, and, conversely, with a few exceptions, outlined above, orthographic boundaries are also token boundaries. As a case in point, in LinES Swedish compounds are not decompounded.

In the ideal case, there are two strong criteria that a link should satisfy. The first relates to meaning: the two halves of a link should be similar in meaning, or, at least, convey roughly the same information to a reader. This rather loose description is motivated by the fact that it is hard to make it in stronger terms. Certainly we cannot generally claim that expressions of two different languages have the same meaning. Moreover, translations are not generally required to be literal or exact, so that translators often deviate from what could be regarded as the closest possible translation. Since those not so faithful translations are among the translations that we would like a user of the treebank to be interested in, we wish to register them. In particular, this criterion allows us to link a common noun to a proper noun, or a pronoun, when they have the same referents.

The second criterion relates to structure. Two segments are assumed to correspond under translation if they have a corresponding external relation to their respective contexts.

The simple cases are when the grammatical structures are identical and the meaning is near synonymous. When this is not the case, we use the following priorities:

- For content words similarity in meaning has preference over structure. Thus, if a noun corresponds to a verb under translation they should be aligned. Similarly, for head switches where a verb may be translated by an adverb, such as *finished packing* ∼ *packat klart* (roughly 'packed ready'), we align *finished* with *klart* and *packing* with *packat*.

- For function words similarity in structure has preference over meaning. Thus, if a function word on one side of the treebank has no correspondent in the form of an independent function word, it is assigned a null link. We also apply restrictions on how function words may correspond at the level of parts-of-speech. Thus, a preposition may correspond to a subjunction, but not to a determiner or a pronoun.

The respect for token boundaries brings with it the difference in the treatment of content words and function words. Since function words of the language pair English-Swedish often corresponds to an affix in the other language, they will have numerous null alignments that are due to grammatical differences between the languages. However, grammatical differences are not the sole cause of null alignments. In particular, if a function word could have been present on the other side, but is not, this is a good argument for assigning it a null alignment rather than as a part of a construction. For example, in English the word *seem* requires an infinitive marker if the complement is a verb phrase, as in *seem to like*, whereas a Swedish counterpart such as *verka* may have the infinity marker, but usually doesn't, as in *verkar (att) tycka om.*

However, in a few cases such as *have to, ought to* the common counterparts in Swedish cannot have an infinitive marker as part of the complement, and, for this reason, these constructions are analysed as complex. Thus we align *have to* ∼ *måste* as a 2-1 alignment. Similarly, compound verbs often employ particles and/or reflexives which we usually prefer to analyse as part of the expression for the meaning, unless the meaning is perfectly compositional. For example, English *I feel tired* is often translated by Sw. compound verb with a reflexive: *Jag känner mig trött.* Since the particular word used in the translation cannot be without the reflexive to express the meaning of the English verb, it is analysed as part of a compound verb, and the link would be of type 1-2.

When two content words correspond structurally, but have non-overlapping meanings, they are assigned null links. For instance, if the noun phrase *a beautiful vase* is translated by *en stor vas* (lit. a big vase), both *beautiful* and *stor* are given null links. On the other hand, if the words have related meanings and thus give the reader roughly the same information in their respective contexts, we align them, even though they need not be synonyms. For example, if English *I will show you* is translated by *Du ska få se* (You will see) we align *show* with *se* and *you* with *Du.*

6.2 General guidelines

Structural correspondence is a matter of degree. We cannot know beforehand which external relations might correspond. However, since the set of functions employed to describe the dependency structure of English and Swedish are equally applicable to both languages, we can define allowable correspondences for them. Some basic cases are covered by the following rules:

- (Equivalence) If e0 and s0 are phrasal heads, that can be aligned on semantic grounds, and e1 and s1 are semantically close dependents, then they can be aligned, irrespective

English token	**Example**	**Swedish translation**	**Mapping**
Auxiliary *do*	He *did* not come	Han Ø kom inte	1-0
Auxiliary *be*	They *were* sleeping	Han Ø sov	1-0
Sw. suffix article	*The* girl laughed	Flicka*n* skrattade	1-0
Definite adjectives	*the* whole team	Ø hela laget	1-0
Possessive reflexives	She shook *her* head	Hon skakade på huvud*et*	1-0
Genitive *of*	the roof *of* the house	huset*s* tak	1-0
Analytic comparison	*more* clever	duktig*are*	1-0

Table 5: Level shifts annotated as null links.

of the type of dependency relation they contract.

- (Isomorphism) If e0 and s0 are corresponding segments, and e1 and s1 have the same dependency relation to e0 and s0, respectively, then they also may correspond. If the semantic criterion is satisfied, we will in fact conclude that they correspond.
- (Added or deleted content) If the meaning of a (content) token is not expressed by any token on the other side, then it is aligned with null.
- (Level shifts) If a token expresses, or marks, a grammatical function, which in the other language is expressed by an affix of a token, then the token is aligned with null. Several applications of this rule are listed in the next section.
- (Convergence or Divergence) If the heads of two corresponding segments have a partial semantic correspondence, and the semantic correspondence may improve by incorporating one or more dependents on either side, they should be so incorporated. A typical case is English compounds such as *prayer book* ∼ *bönbok*.
- (Avoidance of complex mappings) Complex mappings, i.e., $n-m$ mappings where both $n \geq 2$ and $m \geq 2$ are used only if no smaller part of the segments concerned correspond according to other rules.

6.2.1 Examples

This section provides discussion of specific cases of alignments, where the guidelines of the previous section can be applied.

The equivalence guideline can be applied both to clauses and phrasal heads. For clauses it will licence alignment of objects to subjects in the case of an active clause being translated by a passive clause, as in (18), or the alignment of an object with a prepositional complement in the case of different argument structures, as in (19):

(18) *They asked him to leave* ∼ *Han ombads att gådärifrån* (He was asked to leave)
(19) *She trusts him* ∼ *Hon litar påhonom*
(20) *a fascinating story* ∼ *en berättelse som fascinerar* (A stoy that fascinates)

Similarly, in a phrase such as (20) an attribute may be aligned with a modifier.

Isomporhism will apply in a case such as (21), where the assumed identity of reference for the objects is counted as a sufficient semantic similarity. Thus, we would align *Times* ∼ *tidningen*. We would also align *morning* ∼ *morse* but not *this* with *i*, as these have

English token	Example	Swedish translation	Mapping
Infinitive markers	He seems *to* like it	Han verkar Ø tycka om det	1-0
Participial Relatives	a man Ø smoking cigars	en man *som* rökte cigarrer	0-1
Verb sequence	went Ø asking for help	gick *och* bad om hjälp	0-1
Prep. objects	She told Ø me	Hon berättade *för* mig	0-1

Table 6: Added/deleted function words marked as null links.

different functions with respect to their context and thus arguably fall under the conditions for added or deleted content. In principle, the Swedish translation could have used a structurally similar construction such as *denna morgon* and similarly, English could have added a preposition.

(21) *Did you read the Times this morning?* ∼ *Läste du tidningen i morse?*

Table 5 provides examples of level shifts that are treated as null alignments and Table 6 gives examples of other kinds of frequent null alignments in LinES.

6.3 Comparisons with the Blinker guidelines

The word alignment guidelines differ in some respects from other systems, of which the Blinker system (Melamed 1998) is probably the best known and most used. The general recommendation of Blinker, that "you should specify as detailed a correspondence as possible"(p. 8) is certainly adhered to also in LinES, but this very general guideline can be operationalised in many different ways. The reference alignment used in the ARCADE project (Veronis and Langlais 2000) was created under the slogan "Align as small segments as possible, and as long segments as necessary". This is more specific and was applied quite strictly to enforce semantic equivalence of aligned segments. In particular, it handled level shifts by including all the function words necessary to make a segment express the categories that may have been expressed morphologically in the other language. Thus given the pair of sentences *They were sleeping* and its Swedish translation *De sov*, both Blinker and ARCADE would prefer a 2-1 link such as *were sleeping* ∼ *sov*, where LinES uses two smaller links: *were* ∼ Ø, and *sleeping* ∼ *sov*. The main reason for doing so is that the latter option allows for a simple lexically oriented search based on the word alignments, while still enabling search of general structural correspondences, by using the dependency structures. Neither the Blinker project nor ARCADE used parsed sentences.

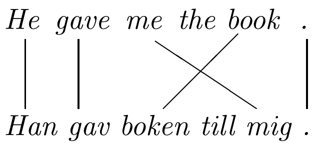
Another difference with the Blinker guidelines concern frozen expressions. The Blinker guidelines says that such phrases should be linked as wholes, especially when they are unique to one or the other language. This rule is illustrated with the French phrase *s'applait* ('call oneself'). However, in a case like this we prefer either that *s'applait* is tokenized as a single token, or, if it is tokenized as two tokens as in the Blinker guidelines, that each token is linked separately. Then, if the segment pair is *her name was* ∼ *elle s' applait*, *name* would only be aligned with *applait*, and *s'* would receive a null link.

The method used in Blinker to register alignments of this kind is to link all tokens that are part of one phrase with all tokens that are part of the other phrase. This way of representing a complex alignment is something we wish to avoid as the LinES way of representing links does not distinguish independent links and links that are merely one part of a larger complex.

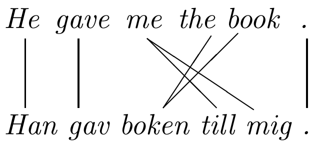
When material is repeated or reduplicated for one language, but not for the other, Blinker recommends that all instances of the repeated unit are linked to the single instance in the other language. This concerns, *inter alia* repeated conjunctions and resumptive pronouns. For instance, given English *Jack, he went home* and a Swedish translation such as simply *Jack gick hem*, Blinker proposes that both tokens *Jack* and *he* on the English side are linked to the Swedish token *Jack*. In LinES, on the other hand, we would use a null link for *he*, especially as Swedish would allow a resumptive pronoun in this context but the translator has chosen not to use one.

Similarly, when prepositional objects correspond to bare objects, Blinker recommends including the preposition in the link, while LinES treat the preposition as extra material that does not have a correspondent in the other language. Note the difference:

LinES:



Blinker:



But, as we stressed above, the Blinker alignment can be obtained as part of the search interface, as a structural correspondence.

References

Lars Ahrenberg. Codified close translation as a standard for MT. In *Proceedings of th The 10th Annual Conference of the European Association for Machine Translation*, Budapest, Hungary, May, 30-31 2005.

Lars Ahrenberg. LinES: An English-Swedish parallel treebank. In *Proceedings of th The 16th Nordic Conference of Computational Linguistics*, Tartu, Estonia, May, 24-26 2007.

Sylvaine Kahane, Alexis Nasr, and Owen Rambow. Pseudo-projectivity: A polynomially parsable non-projective dependency grammar. In *Proceedings of ACL-COLING, 1998*, pages 646–652, Montreal, Canada, 1998.

I. Dan Melamed. Annotation style guide for the Blinker project. *IRCS Technical Report 98-07*, 1998.

Magnus Merkel. *Understanding and enhancing translation by parallel text processing. Linkoping Studies in Science and Technology, Diss. No 607.* PhD thesis, Department of Computer and Information Science, Linköping University, 1999.

Jean Véronis and Philippe Langlais. Evaluation of parallel text alignment systems: The ARCADE project. In Jean Véronis, editor, *Parallel Text Processing: Alignment and Use of Translation Corpora*, pages 369–388. Kluwer Academic Publishers, 2000.