Detecting semantic overlap: A parallel monolingual treebank for Dutch

Erwin Marsi and Emiel Krahmer

Dept. of Communication and Information, Tilburg University

Abstract

This paper describes an ongoing effort to build a large-scale monolingual treebank of parallel/comparable Dutch text, where nodes of syntax trees are aligned and labeled according to a small set of semantic similarity relations. Such a corpus has many potential uses and applications in e.g., multi-document summarization, question-answering and paraphrase extraction. We describe the text material, preprocessing, annotation, and alignment of sentences and syntax trees, both manual and automatic. Two new annotation tools are presented, as well as results from pilot experiments on inter-annotator agreement. On the basis of this resource, new automatic alignment software and NLP applications will be developed.

1 Introduction

We describe an ongoing effort within the context of the DAESO (Detecting And Exploiting Semantic Overlap) project[1] to build a large-scale monolingual treebank of parallel/comparable Dutch text consisting of over 1 million words, where nodes of syntax trees are aligned and labeled according to a small set of semantic similarity relations. This paper aims to attract and inform potential users of this resource, providing information about its design, potential applications, development process, current status, and plans for deriving software tools and NLP applications.

In the first part of the paper we introduce the two key ideas of a parallel monolingual treebank and semantic similarity relations. We discuss potential applications in multi-document summarization, question-answering, information extraction and textual entailment. The remainder of the paper focuses on the construction of the corpus. We describe the raw text material, preprocessing, tokenization and syntactic parsing. Next is alignment at the sentence level, both automatic and manual, and finally alignment of syntax trees. We introduce newly developed annotation tools and present results from pilot experiments on inter-annotator agreement. We finish with a summary and plans for future work on automatic alignment software and NLP applications.

2 Background

2.1 Parallel monolingual treebanks

Treebanks of syntactically annotated sentences have become an essential resource in computational linguistics and related areas. Not only for developing and systematically validating computational models of syntax, but also for data-driven development of natural language processing tools such as part-of-speech taggers,

[1] For the latest developments, see the website at http://daeso.uvt.nl

chunkers and parsers. In a similar vein, large *parallel corpora* of *bilingual* text have become the basis for statistical and example-based machine translation. Typically, the text material in a bilingual parallel corpus is aligned at the level of sentences, words or arbitrary substrings. Several researchers have begun to explore *parallel treebanks* with more linguistically motivated information in the form of aligned phrase-structure trees or dependency structures (see e.g., Gildea (2003) and Samuelsson and Volk (2006)).

A similar type of resource, parallel corpora of *monolingual* text, have proved useful for automatic extraction of synonyms and paraphrases, which in turn has a wide range of applications from machine translation to information retrieval. This has also inspired work on *comparable corpora* of loosely associated text like comparable entries from different encyclopedia (Barzilay and Elhadad 2003).

A logical combination of these two trends – parallel bilingual treebanks on the one hand and monolingual parallel/comparable text corpora on the other – leads to the idea of a *parallel monolingual treebank*, which we define as a corpus of parallel/comparable text in the same language with aligned parse trees. It seems that so far little or no published research has addressed this notion – but see (Ibrahim et al. 2003). In our opinion parallel monolingual treebanks hold great potential, not only for paraphrasing, but also in general for studying the mapping from meaning to alternative surface realizations, and in many NLP applications.

2.2 Tree alignment and semantic similarity relations

Alignment of syntax trees is the process of aligning those pairs of nodes that are similar. More precisely: for each node v in the syntactic structure of a sentence S, its yield $\textsc{str}(v)$ is defined as the substring of all tokens under v (i.e., the composition of the tokens of all nodes reachable from v). An alignment between sentences S and S' then pairs nodes from the syntax trees for both sentences. Aligning node v from the syntax tree T of sentence S with node v' from the tree T' of sentence S' means that there is a similarity relation between their yields $\textsc{str}(v)$ and $\textsc{str}(v')$.

This makes node alignments to some extend similar to the dependencies among words in a syntactic dependency structure. However, where dependencies are normally labeled in terms of a set of meaningful dependency relations, alignments are unlabeled in the majority of work on alignment. If labeled, the set of relations is limited to a binary distinction like *sure* and *possible* (Daumé III and Marcu 2005), or *good* and *fuzzy* (Volk et al. 2006).

In contrast, we propose to label alignments according to a small set of *semantic similarity relations*. By way of example, we use the following pair of Dutch sentences:

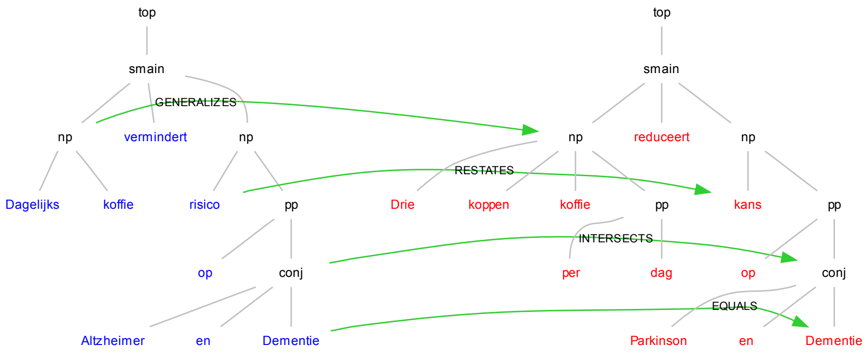


Figure 1: **E**xample of two (partially) aligned syntactic trees

(1) a. Dagelijks koffie vermindert risico op Alzheimer en Dementie.
Daily coffee diminishes risk on Alzheimer and Dementia.

b. Drie koppen koffie per dag reduceert kans op Parkinson en Dementie.
Three cups coffee a day reduces chance on Parkinson and Dementia

The corresponding syntax trees and their (partial) alignment is shown in Figure 1. We distinguish the following five mutually exclusive similarity relations:

1. v **equals** v' iff STR(v) and STR(v') are literally identical (abstracting from case). Example: “Dementie” equals “Dementie”;
2. v **restates** v' iff STR(v) is a paraphrase of STR(v') (same information content but different wording). Example: “risico” restates “kans”;
3. v **generalizes** v' iff STR(v) is more general than STR(v'). Example: “dagelijks koffie” generalizes “drie koppen koffie per dag”;
4. v **specifies** v' iff STR(v) is more specific than STR(v'). Example: “drie koppen koffie per dag” specifies “dagelijks koffie”;
5. v **intersects** v' iff STR(v) and STR(v') share some informational content, but also each express some piece of information not expressed in the other. Example: “Alzheimer en Dementie” intersects “Parkinson en Dementie”

It should be noted that for expository reasons the alignment shown in Figure 1 is not exhaustive.

2.3 Applications

Aligning syntax trees and labeling alignments in terms of these semantic similarity relations has many interesting theoretical and practical implications, some of which are explained below.

Sentence fusion in multi-document summarization Given a set of similar documents, a multi-document summarization system must first identify the most important sentences for inclusion in the summary. To avoid redundancy, the system must be able to detect similar sentences, which amounts to the task of sentence alignment in comparable texts. Summarizers which attempt to produce real summaries – instead of merely extracts – must also revise sentences. On the one hand, they must identify and merge similar information in order to avoid redundancy. On the other hand, they must remove unimportant parts as well as rephrase parts in order to accomplish further sentence compression. One can envision this revision process as one of aligning, merging and pruning syntax trees, followed by the generation of revised sentences using techniques from Natural Language Generation – an approach called *sentence fusion* by Barzilay and McKeown (2005). Our labeled alignments are interesting extension here, because they allow a system to generate fused sentences which are more specific, equivalent or more general than the original ones. Some of our initial work in this area is described in (Marsi and Krahmer 2005a).

Clustering answers in Question-Answering Question-Answering (QA) systems typically analyze a question, search for potential answers in a large body of text material, produce a list of potential answers ranked in order of decreasing likelihood, and show only the topmost answer to the user. For questions of the "open" type, like "What are the risks of overweight?", the topmost answer is unlikely to be optimal. On the one hand, it may be incomplete in the sense that it does not exhaustively list all the risks of overweight encountered in the full text collection. On the other hand, as it is a piece of text extracted from a particular context, it may contain additional information which is irrelevant to the question. We think that detecting and merging similar answers will lead to answers which are both more comprehensive and more to the point. An experiment addressing users' preferences for several different types of answer fusions is described in (Krahmer et al. 2008).

Paraphrases in Information Extraction A large corpus of aligned monolingual text can be used to extract paraphrases from. Here we are interested not so much in synonyms, many of which are already available from other resources like Wordnet, but rather in complex paraphrases as in the pair "X left company Y" and "X, former employee of Y". These may be extracted from aligned syntax trees along the lines of (Lin and Pantel 2001). We plan to investigate to what extent structural paraphrases of this type improve tasks in information retrieval and information extraction.

Recognizing Textual Entailment Recognizing Textual Entailment (RTE) is the task of predicting whether a certain sentence (the hypothesis) is entailed by a one or more other sentences (the text). It is proposed as a general NLP task, akin to Word Sense Disambiguation or Named Entity Recognition, as a common building block for applications like summarization, QA, IR and IE (Dagan et al. 2005). Many RTE systems use a form of alignment under the assumption that entailment is likely if the hypothesis can be aligned with the text. An example of this approach using alignment of syntax trees is described in (Marsi et al. 2006).

3 Corpus material and annotation

3.1 Text material

The corpus contains Dutch text material from five different sources. The target size of the final corpus is 1 million words, half of which is processed with partly manual annotation and correction, whereas the other half will be processed fully automatically. The scope of the current discussion is limited to the first half million.

The composition of the corpus is the result of a trade-off between several constraints. To begin with, we intended to cover the range from true *parallel* text down to loosely associated *comparable text*, preferably across different text genres too. Furthermore, several text types are motivated by potential applications in summarization and QA. Finally, we needed to be able to settle copyright issues in a proper legal manner to ensure that the corpus can be made available to other researchers. This resulted in the following five text components.

Book translations True parallel text was obtained from alternative Dutch translations of foreign language books (about 125k words). Choices for source material were limited to older books, because alternative translations of recent books are extremely hard to come by. The DAESO corpus includes parallel Dutch translations from (parts of) three books: (1) “Le Petit Prince” by Antoine de Saint-Exupéry, (2) “On the Origin of Species” by Charles Darwin (in the 1st and 6th edition) and (3) “Les Essais” by Michel de Montaigne. Admittedly, the latter two books contain dated language use, but we used recent translations in modern Dutch.

Autocue-subtitle pairs This material comes from the *NOS journaal*, the daily news broadcast by the Dutch public television. It consists of the autocue text as read by the news reader and the associated subtitles (125k words). It was tokenized and aligned at the sentence level in the ATRANOS project (Daelemans et al. 2004). Because of space constraints, the subtitles typically present a compressed form of the autocue, making it an excellent source for work on automatic text compression, one of the subtasks in summarization.

News headlines Another source consists of similar headlines from online news articles. These were automatically mined from the Dutch version of Google News (25k words). As the site’s clustering is based on the full article content rather than only the headline, we often found substantial differences between headlines. We therefore performed a manual subclustering in order to eliminate outliers and to extract sets of sufficiently similar sentences.

QA-system output For future work aimed at question-answering, the corpus also contains samples from the QA domain. The IMIX project has developed a multimodal QA system in the medical domain (Theune et al. 2007), where questions are answered by searching a large collection of text ranging from medical encyclopedia to layman websites. In order to evaluate the QA engines, a reference corpus of questions and associated answers as encountered in the available texts was manually compiled. From this corpus, we extracted all clusters of two or more alternative answers. With about 1k words, this segment is relatively small.

Press releases As the main source for comparable text, we used press releases regarding the same news event (225k words). These were obtained from news feeds of ANP and Novum, two Dutch press agencies. This type of material is particularly suitable to bootstrap automatic multi-document summarization. Similar articles were automatically extracted within a certain time window, relying on simple word overlap measures. The automatic procedure aimed at a high recall at the expense of precision, but was followed by manual correction. The reason is that finding similar articles in two large collections is much harder, for humans at least, than deciding whether a given pair of articles is indeed similar.

3.2 Preprocessing

Preprocessing in general involved converting all text material to XML format with UTF-8 character encoding. All book translations were (mostly) automatically converted from their original electronic format (raw text, MS Word, PDF) to XML adhering to the *TEI Lite* standard, the light version of the Text Encoding Initiative markup language (Burnard and Sperberg-McQueen 2006). The original document structure and formatting was preserved as much as possible in the markup; at a minimum, all books have markup indicating chapters and sections. In addition, manual markup was added to indicate parts of the texts which were not fit for our purposes, e.g., citations in a foreign language.

As TEI Lite is not particularly suited for the markup of the other types of text material, these were converted to custom XML formats. For instance, the XML for the headlines stores information regarding timestamps and sources, and indicates clusters and subclusters.

3.3 Tokenization

All text material was subsequently tokenized using the Dutch tokenizer developed within the D-COI project (Martin 2007), with the exception of the autocue-subtitle material, which was already tokenized. Sentence boundaries were indicated by inserting additional sentence tags in the XML source files, where each sentence element was assigned a unique id attribute.

We found that tokenization errors were more frequent in the book material, presumably because of long and complex sentences. As in particular sentence splitting errors will be fatal for the subsequent parsing step, and because tokenization errors are relatively cheap to fix, we undertook manually correction in this corpus segment. Tokenization errors in the press releases where fixed only in so far as they were noticed by the annotators during the subsequent step of sentence alignment.

3.4 Syntactic parsing

Next, the Alpino parser for Dutch (Bouma et al. 2001) was used to parse suitable sentences, excluding e.g., chapter heading, footnotes, citations in a foreign language, etc. The parser provides a relatively theory-neutral syntactic analysis as originally developed for the Spoken Dutch Corpus (van der Wouden et al. 2002). It is a blend of phrase structure analysis and dependency analysis, with a backbone of phrasal constituents and arcs labeled with syntactic function/dependency labels. Output is stored as a *treebank*, which is simply a list of parses in the Alpino XML format, with an id attribute identical to that of the input sentence in the source document.

As no parser is perfect, parsing errors, in the form of partly erroneous or fragmented analyses, are unavoidable. In addition, a few very long and/or particularly ambiguous sentences failed to pass at all. Due to time and cost constraints, such parsing errors were not subject to manual correction.

4 Sentence alignment

The next stage involved aligning similar sentences. Part of the text material was already aligned at the sentence level: the autocue-subtitle segment was aligned within the ATRANOS project; the alternative answers from the QA reference corpus are implicitly aligned, and the same goes for all sentences in subclusters of news headlines. Hence alignment of sentences was required only for the book translations and the press releases. This process was carried out in two steps: automatic alignment followed by manual correction.

4.1 Automatic alignment of parallel translations

Automatic alignment of sentences from parallel translations is a well-studied area for which a number of standard solutions are available, e.g., (Gale and Church 1993). It is usually assumed that the bulk of the alignments is of the 1-to-1 type, and that crossing alignments and unaligned sentences are rare. We found that these assumptions are frequently violated in our samples. For instance, in the two translations of “On the Origin of Species” from different editions, there are many differences, largely due to Darwin’s own revisions. These range from long sentences in one translation being split in multiple sentences in the other to substantial pieces of added or removed text (the 6th edition even has a whole new chapter).

As the initial automatic alignment was known to be manually corrected afterward, we opted for a straightforward pragmatic approach that is robust to above problems. It takes a sentence from the first translation and checks for all sentences in a sliding window over the second translation at approximately the same position whether two sentences are sufficiently similar to justify alignment, where similarity is defined in terms of n-gram overlap. We use a relatively low threshold to get a high recall at the expense of precision, because in practice manually deleting unintended alignments is an easier task than identifying all correct ones.

Obviously, this approach is sensitive to large gaps due to insertion/deletion of substantial pieces of text. We therefore found it beneficial to carry out automatic alignment in multiple passes. That is, first align chapters, next align sections, then paragraphs and finally sentences.

Alignments were stored in a simple custom XML format which contains two types of information: (1) references to the source and target documents; (2) a list of links specifying the tags and id's of the aligned source and target elements respectively. We will refer to an XML document containing this information as a *parallel text corpus*.

4.2 Manual correction of sentence alignments

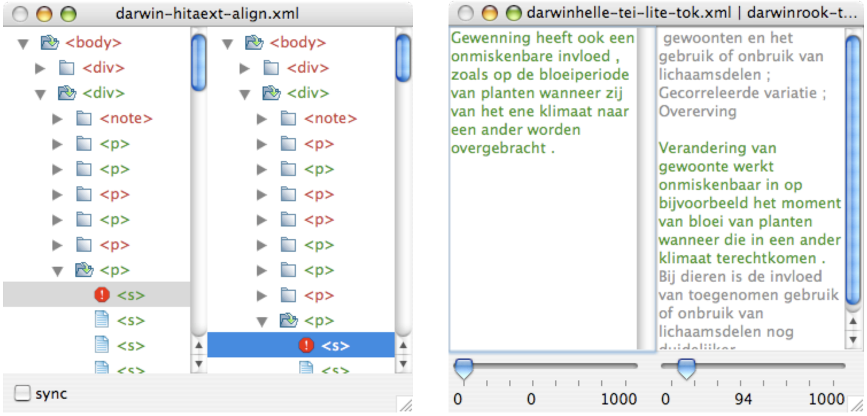
A special alignment annotation tool, called *Hitaext*, was developed for visualizing and editing alignments between text segments.[2] It takes as input a parallel text corpus, i.e. an XML file defining source and target XML documents plus a list of links between elements in those two documents. It then offers two different views on the input documents: the tree view and the text view.

The *tree window* – see the left side of Figure 2 – visualizes the hierarchical structure of the XML elements in the form of a pair of adjacent tree controls (or tree widgets). These allow a user to quickly walk through the document structure using mouse and/or keyboard. In our case, these documents may be TEI Lite XML documents, where the elements correspond to chapters, section, paragraphs and sentences. One of the advantages of this representation is that large documents remain manageable because the user can expand or collapse arbitrary parts of the document tree. In addition, irrelevant elements can be either skipped or hidden completely by configuring the style sheet.

The *text window* – see the right side of Figure 2 – shows the two pieces of text corresponding to the two elements currently in focus (or selected) in the tree window. These are typically the texts of sentences, paragraphs, chapters or even whole documents (when the focus in on the root node). The sliders at the bottom of the text window allow the user to reveal a variable amount of the context.

If an element is aligned, its tag is shown in green in the tree window and its corresponding text is also shown in green in the text window. In contrast, unaligned tags and texts are shown in red. If a focused element has alignments, the aligned elements in the other tree are marked by means of an exclamation mark icon. This representation accommodates 1-to-1, 1-to-n and n-to-m alignments. By selecting two elements from either tree and subsequently hitting the space bar, a user can toggle (switch on/off) the alignment between them.

[2] Hitaext is implemented in wxPython, runs on Mac OS X, Linux and Windows, and is released as open source software from http://daeso.uvt.nl/hitaext



Gewenning heeft ook een onmiskenbare invloed , zoals op de bloeiperiode van planten wanneer zij van het ene klimaat naar een ander worden overgebracht .

gewoonten en het gebruik of onbruik van lichaamsdelen ; Gecorreleerde variatie ; Overerving

Verandering van gewoonte werkt onmiskenbaar in op bijvoorbeeld het moment van bloei van planten wanneer die in een ander klimaat terechtkomen . Bij dieren is de invloed van toegenomen gebruik of onbruik van lichaamsdelen nog duidelijker

Figure 2: Screen shot of Hitaext, the tool used for aligning text segments

One of the distinguishing features of Hitaext in comparison with other text alignment tools is its support for simultaneous alignment at multiple annotation levels (e.g., words, sentences, paragraphs, and chapters). Another feature is the lack of a predefined input format for the source and target documents; the only constraint is that the XML must be well-formed. In fact, this makes Hitaext a general graphical tool for aligning text elements in pairs of arbitrary XML documents. It might therefore as well be used for tasks like aligning bilingual text at the word level or aligning ontological hierarchies. Other features include automatic synchronization of the two trees, i.e. focus automatically jumping to the (first) aligned element in the other tree, and quickly walking through all aligned elements in the case of one-to-many or many-to-many alignment.

4.3 Alignment of comparable text

The assumptions about parallel text clearly do not hold in the case of comparable text: one-to-many alignments – or even many-to-many – are to be expected frequently, just as crossing alignments and large portions of unaligned material. Moreover, similarity between sentences, and therefore the decision to align or not, is much more gradient. Whereas with parallel translations it is virtually always evident whether or not two sentences are translations of the same source sentence, it is much harder to decide whether two comparable sentences are sufficiently similar to justify alignment. In order to preserve a reasonable level of consistency among annotators, we have developed a set of annotation guidelines.

For a start, there is no need for aligned sentences to be true paraphrases of each other. One sentence may contain pieces of information which are not present in the other. Likewise, information in one sentence may be more specific/general than in the other. However, aligned sentences should have at least one proposition in common, which we interpret informally as a statement about someone or something.

Examples of (partial) sentences including the same proposition:

- Balkenende is the minister-president of the Netherlands
- Balkenende, who is the minister-president of the Netherlands, ...
- Balkenende, the minister-president of the Netherlands ...
- Balkenende as the minister-president of the Netherlands ...
- Balkenende being minister-president of the Netherlands ...

However, the following examples do not share a proposition (even though they may share some words):

- Balkenende is a wine expert
- Balkenende likes to barbecue
- Bush likes to barbecue
- the minister-president of the Netherlands
- the capital of the Netherlands

Furthermore, we do not attempt to align each sentence with *the* most similar sentence (one-to-one alignment). Instead, we align each sentence to every other sentence with which it has sufficient overlap, i.e. at least one proposition in common, effectively allowing many-to-many alignment.

We also allow use of common sense. Consider the following pair:

- Keith Urban left US rehabilitation clinic
- Keith Urban cured from addiction

In the strict logical sense, these statements differ: in theory, one may leave the clinic without being cured, or one may be cured but remain in the clinic. However, in the context of two texts on the same topic, we prefer to view them as similar for all practical purposes. This is in the same spirit as *natural entailment* is defined in the Recognizing Textual Entailment task (Dagan et al. 2005). Other examples include approximately identical locations, quantities, times, etc. In a similar vein, referring expressions like pronouns or generic definite descriptions may be interpreted in the context.

However, we refrain from alignment in cases where inferring similarity requires elaborate reasoning and background knowledge, as in:

- The Radicals now hold 80 of the 250 seats in parliament
- The SRS is currently the biggest party in Serbia

Notice that this would requires one to know that 80 out of 250 is a majority because all other political parties hold fewer seats.

4.4 Inter-annotator agreement

We carried out a pilot experiment with two annotators who each aligned the same 10 pairs of comparable press releases, varying in length from 4 tot 33 sentences. The total number of possible one-to-one sentence alignments to consider was 1492. Both annotators agreed on 44 but disagreed on 32 alignments. While discussing the differences, we encountered some difficult cases which gave rise to revision of the annotation guidelines. However, it turned out that the majority of the disagreements were caused because an annotator simply failed to notice a particular alignment.

After revision of the guidelines, we repeated the experiment with another set of 10 comparable press releases, this time with 1337 possible alignments to consider. One annotator made 15 unique alignments and the other 39, in other words, they disagreed on 54 cases, while both agreed on 51 alignments. Even though the ratio is slightly worse than before, this time by far the most disagreements were caused by overlooked alignments. This supports our impression that the task of identifying *all* pairs of similar sentences is harder than deciding on similarity of a particular pair of sentences. In terms of precision and recall, this means that precision on sentence alignment is substantially better than recall.

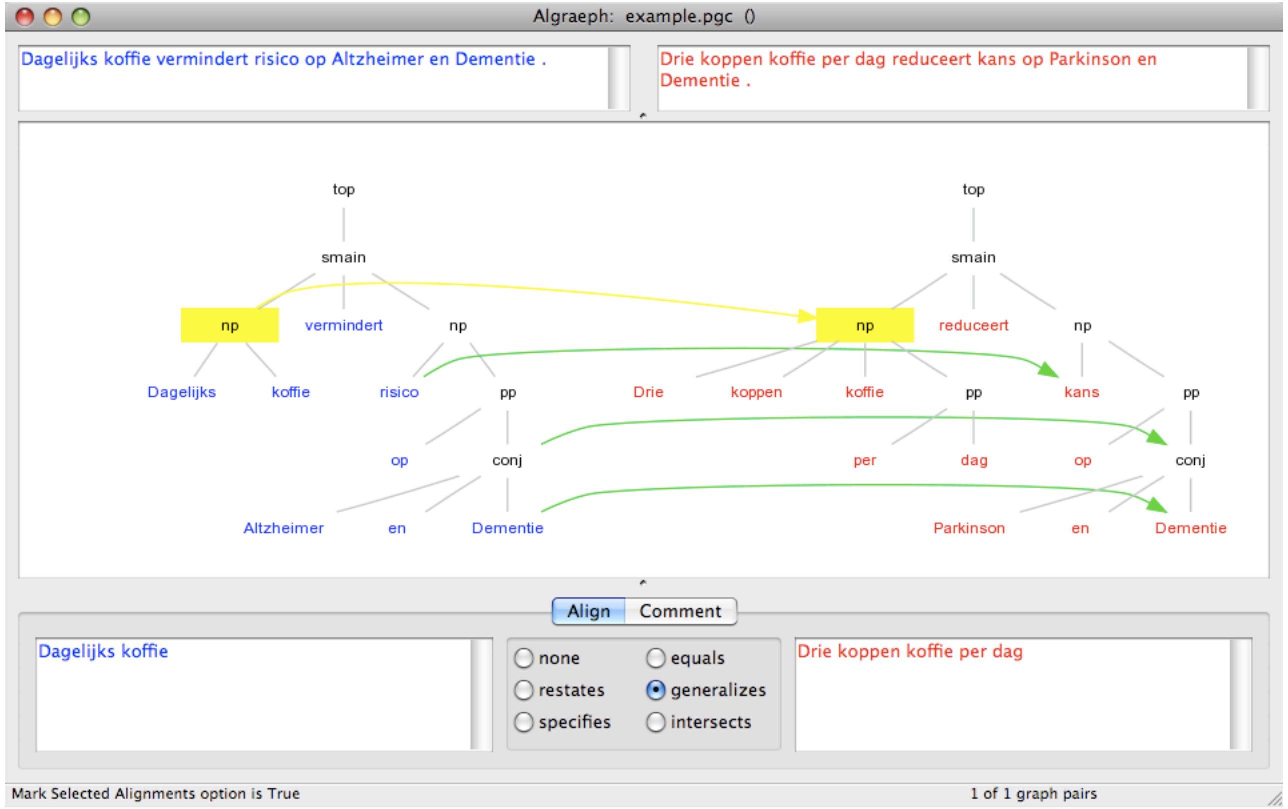
5 Syntax tree alignment

The final stage consists of aligning the syntax trees according to the semantic alignment relations described in Section 2.2. For creating and labeling alignments, we developed another special-purpose annotation tool called *Algraeph*.[3] The screen shot in Figure 3 shows Algraeph with the sentence pair from example (1). The input sentences are shown in the text boxes at the top. The corresponding syntactic trees are shown in the middle, with alignments indicated by colored lines. The focused nodes and their alignments are shown in yellow. The token sequences corresponding to these focused nodes are shown in the text boxes at the bottom, with the alignment relation, which is ”generalizes” here, in between. This relation can be changed or removed by clicking any of the radio buttons.

One of the challenges for annotators is dealing with the large syntax trees of long sentences, which are hard to navigate. Algraeph therefore has a range of options to tune the rendering of the trees and alignments.

Input to Algraeph is a *parallel graph corpus*, a file in a simple custom XML format which contains three types of information. First, one or more references to the treebanks containing the syntactic trees of the aligned sentences. Second, a list of linked trees which are identified by the id’s of the source and target trees and the id’s of the treebanks they originate from. Third, for each pair of linked trees, a list of linked nodes in terms of the id’s of the aligned source and target nodes as well as the alignment relation.

[3] Algraeph is a rewritten and extended version of our earlier tool called Gadget. It is implemented in wxPython, runs on Mac OS X, Linux and Windows, and is available as open source software from http://daeso.uvt.nl/algraeph



Algraeph: example.pgc ()

Dagelijks koffie vermindert risico op Altzheimer en Dementie .

Drie koppen koffie per dag reduceert kans op Parkinson en Dementie .

Dagelijks koffie

○ none ○ equals
○ restates ● generalizes
○ specifies ○ intersects

Drie koppen koffie per dag

Mark Selected Alignments option is True

1 of 1 graph pairs

Figure 3: Screen shot of Algraeph, the tool used for aligning syntax trees

A parallel graph corpus is automatically derived from the combination of a source and target text document, a parallel text corpus, and the corresponding treebanks. In order to speed up the manual annotation, alignments of the type "equals" are established automatically, as these can be be reliably detected by automatic procedures. Our aim is to bootstrap the annotation process so other alignment relation can also be automatically predicted and the manual annotation process will ultimately involve only correction work.

5.1 Inter-annotator agreement

As is to be expected, the correct alignment is not always evident. For example, because of differences in interpreting the meaning of a particular word or phrase, or because of blurry edges between the categories of semantic relations. Other issues arose because of parsing errors which prevent one from making the desired alignment. We are currently in the process of writing an annotation manual with guidelines to resolve these issues in a consistent manner. In general we adhere to the same principles we discussed earlier in the context of alignment of comparable sentences. For instance, referring expressions may be interpreted in the context and the use of common sense is allowed.

In (Marsi & Krahmer 2005a) we reported on a pilot experiment which involved aligning syntax trees from the first five chapters of "Le Petit Prince". Results indicated that humans can perform this task well, with ultimately an F-score of .98 on creating alignments and an F-score of .95 on assigning semantic similarity relations. We also presented results on automatic annotation, which achieved an F-score on alignment of .85 and an F-score of .80 on semantic relation classification, albeit with some strong assumptions regarding prior knowledge.

We repeated this experiment on a smaller scale with 30 sentence pairs from the same book translations, but this time with other annotators. Again the F-score on assigning semantic similarity relations was over .95. It should be noted though that due to the nature of the text material the majority of the alignments involved "equals", and therefore the agreement is relatively high. We expect lower figures when we repeat the experiment with other types of text material in the corpus. At any rate, the current corpus will allow us to repeat these experiments on larger scale and with more challenging text material in the near future.

6 Summary and future work

We have introduced the idea of parallel monolingual treebanks and of alignments labeled according to a set of five semantic similarity relations. We have suggested potential applications in multi-document summarization, QA, IE and RTE. Next, we described the ongoing effort to build a large-scale parallel/comparable monolingual treebank for Dutch of over 1 million words. We described the text material from different sources (book translation, autocue-subtitles, news headlines, answers from the QA domain, and press releases) and the process of preprocessing, tokenizing, and syntactic parsing. Next, we addressed alignment at the sentence level, both automatic and manual, followed by alignment at the syntax level. Two new graphical annotation tools were presented (Hitaext and Algraeph). Results from pilot experiments on inter-annotator agreements showed encouraging results. We are currently working on the manual alignment of syntax trees for the first half million words of the corpus.

Apart from the two graphical tools for manual alignment, we are developing software for automatic alignment of sentences from parallel and comparable text sources. Likewise, we will continue work on automatic alignment of syntax trees along the lines of (Marsi and Krahmer 2005b). Evidently, the corpus is an excellent resource for this. Once reasonably reliable tools for alignment are in place, we intend to double the size of our corpus to 1 million words by automatically aligning more text material, from the same sources in roughly equal proportions.

In addition, we intend to exploit our tools for detecting semantic overlap in a number of practical applications. In the context of multi-document summarization, we intend to take advantage of the semantic labeling of the alignments, which allows us to generate fused sentences which are more specific, equivalent or more general than the original ones. Some of our initial work in this area is described in (Marsi & Krahmer 2005a). The corpus segment containing QA answers will be used to bootstrap work on automatic clustering and fusing of similar answers from QA systems. Finally, we plan to measure the contribution of automatically derived structural paraphrases in IR and IE.

Acknowledgments We would like to thank Nienke Eckhardt, Paul van Pelt, Hanneke Schoormans and Jurry de Vos for all the annotation work, Erik Tsjong Kim Sang and colleagues for the autocue-subtitle material from the ATRANOS project, Gosse Bouma and others from the IMIX project for the QA reference corpus, and Wauter Bosma for mining the headlines from Google News. We also like to express our gratitude to the publishers and press agencies for providing the raw text material. This work was conducted within the DAESO project funded by the Stevin program (De Nederlandse Taalunie).

References

Barzilay, R. and K.R. McKeown (2005), Sentence fusion for multidocument news summarization, *Computational Linguistics* **31**(3), 297–328.

Barzilay, R. and N. Elhadad (2003), Sentence alignment for monolingual comparable corpora, *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pp. 25–32.

Bouma, G., G. van Noord, and R. Malouf (2001), Alpino: Wide-coverage computational analysis of Dutch, *Computational Linguistics in the Netherlands 2000*, Rodopi, Amsterdam, New York, pp. 45–59.

Burnard, L. and C. M. Sperberg-McQueen (2006), TEI Lite: Encoding for interchange: an introduction to the TEI Revised for TEI P5 release, *Technical report*, Text Encoding Initiative.

Daelemans, W., A. Höthker, and E. Tjong Kim Sang (2004), Automatic sentence simplification for subtitling in Dutch and English, *Proceedings of the 4th International Conference on Language Resources and Evaluation*, pp. 1045–1048.

Dagan, I., O. Glickman, and B. Magnini (2005), The PASCAL Recognising Textual Entailment Challenge, *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*, Southampton, U.K.

Daumé III, H. and D. Marcu (2005), Induction of word and phrase alignments for automatic document summarization, *Computational Linguistics* **31**(4), 505–530.

Gale, W.A. and K.W. Church (1993), A program for aligning sentences in bilingual corpora, *Computational Linguistics* **19**(1), 75–102.

Gildea, D. (2003), Loosely tree-based alignment for machine translation, *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Sapporo, Japan, pp. 80–87.

Ibrahim, A., B. Katz, and J. Lin (2003), Extracting structural paraphrases from aligned monolingual corpora, *Proceedings of the second international workshop on Paraphrasing*, Vol. 16, ACL, Sapporo, Japan, pp. 57–64.

Krahmer, E., E. Marsi, and P. van Pelt (2008), Query-based sentence fusion is better defined and leads to more preferred results than generic sentence fusion, *The 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Columbus, Ohio, pp. 193–196.

Lin, D. and P. Pantel (2001), Discovery of inference rules for question answering, *Natural Language Engineering* **7**(4), 343–360.

Marsi, E. and E. Krahmer (2005a), Explorations in sentence fusion, *Proceedings of the 10th European Workshop on Natural Language Generation*, Aberdeen, GB, pp. 109–117.

Marsi, E. and E. Krahmer (2005b), Semantic classification by humans and machines, *ACL 2005 Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, Ann Arbor, Michigan, pp. 1–6.

Marsi, E., E. Krahmer, W. Bosma, and M. Theune (2006), Normalized alignment of dependency trees for detecting textual entailment, *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, pp. 56–61.

Martin, R. (2007), Sentence-splitting and tokenization in D-Coi, *Technical Report 07-07*, ILK Research Group.

Samuelsson, Y. and M. Volk (2006), Phrase alignment in parallel treebanks, *Proceedings of 5th Workshop on Treebanks and Linguistic Theories*, Prague, Czech Republik.

Theune, M., B. van Schooten, R. op den Akker, W. Bosma, D. Hofs, A.Nijholt, E. Krahmer, C. van Hooijdonk, and E. Marsi (2007), Questions, pictures, answers: Introducing pictures in question-answering systems, *ACTAS-1 of X Symposio Internacional de Comunicacion Social*, Santiago de Cuba, pp. 450–463.

van der Wouden, T., H. Hoekstra, M. Moortgat, B. Renmans, and I. Schuurman (2002), Syntactic analysis in the Spoken Dutch Corpus, *Proceedings of the third International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, Spain, pp. 768–773.

Volk, M., S. Gustafson-Capkova, J. Lundborg, T. Marek, Y. Samuelsson, and F. Tidstrom (2006), XML-based phrase alignment in parallel treebanks, *Proceedings of EACL Workshop on Multidimensional Markup in Natural Language Processing*, Trento, Italy, pp. 93–96.