8

Evaluating deep syntactic parsing

Using TOSCA for the analysis of why-questions

Daphne Theijssen, Suzan Verberne, Nelleke Oostdijk, and Lou Boves
Radboud University Nijmegen

Abstract

Previous research has shown that the high level of detail in syntactic trees produced by the TOSCA parsing system (Oostdijk 1996) is beneficial to *why*-question answering (QA) (Verberne et al. 2006b). TOSCA is an interactive system, i.e. it needs human verification after automatic tagging and parsing. Since only manually corrected TOSCA output has been offered to the *why*-QA system until now, TOSCA needs extrinsic evaluation of its use in the *why*-QA system. In this paper we present a necessary step towards it, namely an intrinsic evaluation of the performance of TOSCA on *why*-questions, which also enables us to trace elements in the parser that leave room for improvement. The evaluation shows that the modularity of the current TOSCA system has a dramatic effect on its performance: Tagging errors and missing syntactic markers radically decrease the coverage and the Parseval scores. Applying the Leaf-Ancestor Assessment metric for parser evaluation, we conclude that the level of detail does not really affect parser accuracy. This stimulates the automatic use of the parsing component in TOSCA for the purpose of *why*-QA. A new version of TOSCA is under construction, in which the level of detail in the parses is maintained, while there is no longer a need to separately provide POS tags or insert any syntactic markers.

Proceedings of the 17th Meeting of Computational Linguistics in the Netherlands
Edited by: Peter Dirix, Ineke Schuurman, Vincent Vandeghinste, and Frank Van Eynde.

8.1 Introduction

In the field of computational linguistics parsers have been developed for generating syntactic analyses. Evaluating parser performance is useful for locating elements of the parser that leave room for improvement. If a parser is not applied for the purpose of a language technology application, evaluation is typically intrinsic, i.e. measuring the performance of a parser in the framework it is created in by comparing parser output to a truth at the right hand, a gold standard. This type of evaluation differs greatly from extrinsic evaluation, where the benefit of the parser to a language technological application is established. In this paper, we undertake an intrinsic evaluation of (the performance of) a parsing system designed for linguistic purposes – more specifically for the linguistic annotation of text corpora – that is being employed in a *why*-question answering system (Verberne et al. 2006b). In doing so, we can (1) facilitate extrinsic evaluation of the parsing system in the context of *why*-QA, and (2) formulate suggestions for a future version.

```
NOFU,TXTU ()
├── UTT,S (act,indic,intens,inter,pres,unm)
│   ├── A,AVP (inter)
│   │   └── AVHD,ADV (inter) – Why
│   ├── INTOP,LV (indic,intens,pres) – are
│   ├── SU,NP ()
│   │   ├── DT,DTP ()
│   │   │   └── DTCE,ART (def) – the
│   │   ├── NPHD,N (com,plu) – attorneys
│   │   └── NPPO,PP ()
│   │       ├── P,PREP () – for
│   │       └── PC,NP ()
│   │           ├── DT,DTP ()
│   │           │   └── DTCE,ART – (def) – the
│   │           └── NPHD,N (com,sing) – Bush administration
│   └── CS,AJP (prd)
│       ├── AJHD,ADJ (prd) – present
│       └── AJPO,PP ()
│           ├── P,PREP () – at
│           └── PC,NP ()
│               ├── DT,DTP ()
│               │   └── DTCE,ART (def) – the
│               └── NPHD,N (com,sing) – hearing
└── PUNC,PM (inter,qm) – ?
```

Figure 8.1: Example of TOSCA output for the question *Why are the attorneys for the Bush administration present at the hearing?*

The parser examined in the present study is the TOSCA system (Oostdijk 1996), an interactive syntactic parser that yields very detailed analyses of English text. An example of a syntactic analysis by TOSCA is presented in Figure 8.1.

TOSCA analyses are constituency trees and essentially include three types of information: (1) information pertaining to the categorial realization of constituents (e.g. article, noun phrase, clause), (2) information about the functional role of a constituent (e.g. prepositional complement, subject) and (3) additional information (for example about the word order observed or the subclass of particular word class) which is presented in the form of attributes. The three levels of analysis interfere with each other: Incorrect categories and/or attributes may lead to the erroneous assignment of function labels to constituents, for instance because the distribution of thematic roles (e.g. direct object) depends on verb transitivity.

The TOSCA system consists of two automatic components, being a part-of-speech (POS) tagger and a parser. In the current paper, the terms ‘tagger’ and ‘parser’ are only used to indicate these particular automatic components, while the total system in which both are embedded is referred to as ‘TOSCA’ or the ‘(TOSCA) system’. Input to the system are text inputs that usually take the form of sentences. These are tagged automatically with part-of-speech (POS) information. The POS tagger is probabilistic and has been trained on a manually annotated corpus. The probabilities are based on the frequency of observed word classes and the immediate context (trigrams) of each individual token in the corpus. The tag set is elaborate; it includes the basic word classes such as ‘article’, ‘preposition’, ‘noun’, etc., but also further subclassifications for most word classes. ’Verbs’, for example, can be subdivided according to their complementation type (transitivity) and form (tense, mode and number) (van Halteren and Oostdijk 1993). The human analyst working with the system verifies whether with each of the tokens in the input string the correct tag is associated. Moreover, where required, the analyst inserts syntactic markers that help reduce the degree of ambiguity of highly ambiguous strings such as prepositional phrases and coordinated constituents. The unambiguously tagged input along with the syntactic markers that have been added is then submitted to the parser. The parser is rule-based and the formal grammar underlying it is based on the descriptive system proposed by Aarts and Aarts (1982), which is an adaptation of the English grammar by Quirk et al. (1972). Erroneously selected POS tags greatly influence the range of possible syntactic structures that can be yielded by the parser. The monotransitive form of *decline* in *Why did the Cincinnati Public schools decline to carry the program?*, for example, might be incorrectly tagged as an intransitive verb. Consequently, the clause *to carry the program* cannot be classified in any of the available syntactic structures because the verb attribute ‘intransitive’ prevents the assignment of the correct function to this direct object. Since the parser has no knowledge of the contextual (i.e semantic, pragmatic and extra-linguistic) knowledge that is called upon, it generates all possible syntactic analyses. However, it includes a penalty system that favours certain intuitively more appropriate analyses than others. It prefers, for example, unmarked word order over marked word order. Still a number of parses with equal penalties may remain, from which the human analyst is expected to select the one correct analysis for storage in a linguistic database. For more details on the TOSCA system, the reader is referred to van Halteren and Oostdijk (1993).

Previous research has already indicated that the deep linguistic information provided by the TOSCA parses is useful for the *why*-question answering system currently under construction at the Radboud University Nijmegen (Verberne et al. 2006b). Until now only manually corrected TOSCA output has been offered to the system. The intrinsic evaluation presented here is the first step towards a necessary extrinsic evaluation of the use of TOSCA in the *why*-QA system. Consequently, the data used consists of *why*-questions solely. For the purpose of discovering items open to improvement, the pipelined design of the TOSCA system and the descriptive model of the parser need to be evaluated. Therefore, the aim of the present study is two-fold:

1. to evaluate the separate stages in the analysis process (POS tagging, tag selection and marker insertion, parsing and parse selection) and the way in which errors in one stage affect subsequent stages;

2. to evaluate the descriptive model used by the TOSCA grammar (incl. categories, functions and attributes and the interaction between these).

The structure of this article is as follows: The evaluation of the separate stages used in arriving at the contextually appropriate analysis for a given string is presented in section 8.2. Section 8.3 concerns the evaluation of the descriptive model of the grammar underlying the parser. Section 8.4 contains our overall conclusion and suggestions for future research.

8.2 Evaluation of separate parser modules

In the introduction the TOSCA system has been described as an interactive system. In this section we investigate the performance obtained in the different stages in the analysis process: (1) automatic tagging, (2) manual tag correction and syntactic marker insertion, (3) automatic parsing and (4) manual parse selection. To establish the effect of inaccuracies on subsequent steps, we skip over the stages requiring human intervention and evaluate the eventual parser output.

8.2.1 Data

As mentioned in the introduction, the data set consists of *why*-questions solely. *Why*-questions can be defined as interrogative sentences with the interrogative adverb *why* or one of its synonyms in (near) initial position. Despite the fact that several data sets have been developed for the purpose of question answering (QA), none of them was suitable for developing and testing a system for *why*-QA. Therefore, Verberne et al. (2006a) developed a data set by asking native speakers of English to formulate *why*-questions to thirteen different newspaper texts, with the explicit mention that the answer to the question should be present in the text. We decided to use a subset of these data for the evaluation of TOSCA. Of the first six texts we included all 138 unique questions in our data set, supplemented with another 100 questions randomly selected from the other seven texts, thus leading to a data set consisting of 238 questions. It was not feasible to use all available *why*-questions because creating gold standard parse trees is very time consuming.

For the purpose of evaluating the separate contributions of the system's components, we derived three data sets from the 238 questions: (1) a gold standard (from now on referred to as 'GOLD'), (2) a semi-automatic output ('SEMI'), in which we applied tag correction and manual insertion of syntactic markers, and (3) a fully automatic output ('AUTO'), in which only the two automatic components (POS tagger and parser) are used. Using the interactive TOSCA system we developed GOLD. For questions that could not be parsed despite our intervention after the tagging and parsing stages, we manually created gold standard trees. SEMI has been obtained by employing the interactive TOSCA system as it was meant up until the actual parsing process. Often, the parser proposed more than one possible syntactic analysis. The order in which these parses are presented is not based on linguistic theory but depends on the system's procedure of passing through the grammatical rules. For SEMI, we always saved the first proposed tree, which is neither ranked first nor completely randomly selected by the parser. To create AUTO, the list of tags proposed by the POS tagger and the first tree proposed by the parser were left unchanged. In this set-up no syntactic markers are inserted because this would involve changing the system (the insertion of syntactic markers presently requires manual intervention on the part of the human analyst; the alternative of producing a script that guesses the location of the markers would be possible, but would alter the system).

8.2.2 Method

SEMI and AUTO can be used for evaluation of the separate stages in the analysis process. An investigation of the outputs obtained while having the system operate fully automatically enables us to establish the effect that omitting tag correction and marker insertion, and refraining from parse selection (ranking) has on the eventual parser output, or put differently, the implications of corruptions in the parser input on parser performance.

We evaluate both the coverage of the parser and the quality of its output. In order to measure the robustness of the parser, we calculate the proportion of questions for which the parser was able to produce output (the coverage) for both SEMI and AUTO. We then try to find explanations for uncovered questions. In order to measure the quality of the parser output, we use Parseval, which is the common metric for evaluation of the quality of constituency trees. Parseval is also referred to as GEIG (Black et al. 1991). Parseval's evaluation method is based on lining up the brackets delimiting constituents. A sentence *a b c* with a gold standard *[a b] c* for instance, is considered not structurally consistent with an output *a [b c]*, because there is a crossing error (Black 1993). In addition to the average number of crossing brackets, precision and recall are calculated. The precision is a ratio of the number of correct brackets in the system's parse to the total number of brackets in the system's parse, while the recall is a ratio of the number of correct brackets in the system's parse to the total number of brackets in the gold standard. Following van Rijsbergen (1979), the F-score can be calculated, which represents the harmonic mean of precision and recall. Since Black (1993), the Parseval metric has been extended. Magerman (1995) has decided to include the assignment of labels in the metric. For example, if the gold standard is *[PP [P a] [NP b]] [VP c]* and the parser output *[ADV a] [VP [V b] [V c]]*, the evaluation is based on comparisons of the location of the brackets as well as the choice of labels. This has led to the measures ‘labelled precision’ and ‘labelled recall’.

Drawbacks of the Parseval metric are that it tends to favour minimal structure (Carroll et al. 1998) and that misattachments are penalised more than once (Lin 1995). The former can be explained by the fact that the more brackets there are, the more errors can be made. For the latter the reader is referred to Lin’s (1995) example on PP-attachment, where a single error is penalised three times. The objections to Parseval have led to the development of various depencency-based parse evaluation methods (e.g. Lin 1995, Carroll et al. 1998). Since TOSCA is a constituency-based parser, the TOSCA output would have to be transformed into a uniform format convenient for the method used if we decided to use a dependency-based evaluation method. This would increase the risk of making errors and thereby decrease the performance reached. Furthermore, most of the rich syntactic information provided by TOSCA will be lost in the transformation, while we intend to use an evaluation method capable of dealing with the high degree of detail in the trees. Therefore, dependency-based methods are not suitable for the present evaluation. Fortunately, the Parseval metric has benefits that justify its use for the present evaluation, namely the fact that it is commonly employed in parser evaluation and that it enables dealing with the three types of information provided by TOSCA, as previously mentioned: categories, functions, and attributes. Following Parseval, we are able to determine the average number of crossing brackets and the labelled precision, recall and F-score for all three types separately, and average them. Averaging seems the best method to get to a single score, because multiplying the scores would penalise related errors more than once. By comparing the scores obtained for SEMI and AUTO, we can draw conclusions on the influence of errors in separate components on the eventual system output.

8.2.3 Results

The coverage, the number of perfect matches and the Parseval scores are presented in Table 8.1. From the set of 238 questions, TOSCA was able to parse 233 in SEMI, and only 190 in AUTO. Of 233 questions in SEMI, 188 were a perfect match to GOLD, compared to only 41 of 190 trees in AUTO. AUTO achieves a lower precision and recall and has more crossing brackets than SEMI (the differences in Parseval scores are significant (p=0.000) following the independent t-test). In AUTO, 84.5% of the POS tags including their specifications (e.g. *V(intr, inf)*) is completely correct for this data set.

8.2.4 Discussion

The differences between SEMI and AUTO in Table 8.1 demonstrate that the accuracy of the tags provided to the parser is essential for the performance of the

Table 8.1: Tag accuracy, coverage, perfect match and Parseval scores for SEMI and AUTO

	SEMI	AUTO
Tag accuracy	1.000	0.845
Coverage	0.979 (233 of 238)	0.798 (190 of 238)
Perfect match	0.807 (188 of 233)	0.216 (41 of 190)
Labelled Precision	0.960	0.794
Labelled Recall	0.957	0.772
Labelled F-value	0.959	0.783
Average nr crossing brackets	0.060	0.310

TOSCA system. This is obvious since the parser is designed so as to produce (minimally) the correct parse on the basis of correctly tagged input. Erroneously tagged input will cause the parser to fail to produce a correct parse. Thus human intervention is required to manually correct any erroneous tags resulting from the application of the POS tagger.

In more than 80% of the covered questions in SEMI, there is no need for the human analyser to select the correct syntactic tree, since it is presented first (0.807 perfect match). Taking into account the fact that the parser does not include a ranking procedure for trees that have obtained equal penalties during the parsing process, we consider this percentage of perfect matches rather large. It encourages a fully automatic use of the parser (i.e. the second automatic component of the TOSCA system) for the purpose of *why*-question answering.

Despite the fact that the parser is a wide-coverage parser intended to parse unrestricted input, we found that for 5 questions in our data set it was unable to produce an analysis, even when provided with gold standard tags (SEMI). Two questions included a coordination that apparently was too complex, another two were problematic because of the percent symbol (%) and one question included a date (*April 26, 1990*). For AUTO, the same problems occurred, except for the last-mentioned, where a tree could be produced due to tagging errors. However, in AUTO another 44 questions could not be parsed:

1. In 24 questions (54.5% of 44 uncovered questions), the proposed POS-tags caused problems with the verb phrase. In some cases, the lexical verb was not tagged as such and therefore was regarded missing by the parser. For example, in the question *Why did hundreds of thousands of people* ***march*** *in Washington twice this year?*), the lexical verb *march* was erroneously tagged as a noun. This leads to serious complications in function assignment. Other problems in the verb phrase concerned the lack or surplus of finite verbs and the inconsistency between the auxiliary and the tense of the lexical verb.

2. The lack of syntactic markers caused problems with coordination in 9 questions (20.5%), for example *Why is the decision expected by late June* ***or*** *early July?*, where the coordinated elements *late June* and *early July* were not recognised as such by the parser because of the missing marks.

3. In 8 questions (18.2%) there were problems with arguments and complements that were not caused by the incorrect tagging of verbs. These cases included instances where nouns were tagged as adverbs or adjectives causing problems in subjects and in prepositional phrases. Moreover, in some cases, a word was incorrectly tagged as a subordinating conjunction, expecting a clause while there was none, as in *why don't they like* ***that*** *idea?*

4. In 3 questions there were problems with existential *there* (it was tagged as a general adverb which was not possible at that location given the context), for instance in *why is* ***there*** *resistance to the Classroom Channel?*

The Parseval scores in Table 8.1 are significantly lower for AUTO than for SEMI, meaning that the TOSCA analyses in this case are more erroneous. Taking into consideration this finding and also the coverage, we can conclude that the parser can only perform well if it is provided with accurate input. The parser is not very robust in handling tagging errors and missing markers. As we observed above, the parser will definitely fail to produce the correct analysis if provided with incorrect or incomplete input, while in some cases there will be no output at all. Thus inaccurate input is always fatal when it comes to parse selection/ranking.

For this particular data set consisting of *why*-questions only, the accuracy of the input could be improved by training the (probabilistic) tagger on a large corpus of *why*-questions, and guessing syntactic markers by use of a script. The benefit of such solutions, however, depends on the size and uniformity of the data set concerned. It is worthwhile to establish whether a different design of the parser performs better, for example an integrated system in which the parser operates on raw input and has direct access to a lexicon, rather than a highly modularised system where POS tagging and tag selection are separate steps which are executed independently of the parser. In such a design there would be no need for human intervention since the parser would be able to negotiate the correct word class tags for the tokens in the input all by itself. Presently, such a system is being developed. The new TOSCA system is still designed to produce syntactic annotations for unrestricted (correct) English which should include the one contextually appropriate analysis for a given input string. Since more than one analysis may be produced by the parser, the system also includes a selection tool which the human analyst can use to make the appropriate selection.

8.3 Evaluating the descriptive model: categories, functions and attributes

As mentioned earlier, the TOSCA parser produces detailed syntactic analyses, indicating categories, functions and attributes. In this section we investigate the parser accuracy on all three types of labels, taking into consideration that the types are interrelated. For example, if the transitivity associated with the verb is incorrect, the subsequent assignment of syntactic roles is bound to be problematic (the parser will either fail completely or at least fail to assign the correct function labels). Investigating how accurate the parser is with each of the types of information helps us in establishing whether the level of detail of the parser output does not lead to more complications than benefits. In this way, we are able to evaluate the descriptive model of the grammar underlying the TOSCA parser.

8.3.1 Data and method

For the evaluation of the different levels of information produced by the TOSCA parser we use the SEMI data we created for the evaluation of the pipelined design of the whole TOSCA system in the previous section. This data set consists of the 233 questions for which the parser was able to produce output. Moreover, we reuse the gold standard (GOLD) we have already developed.

The Parseval metric applied in section 8.2 provides us with several quality scores for each question, but is not helpful in pinpointing where exactly the errors are made. Therefore, we employ the approach proposed by Sampson et al. (1989), and further discussed by Sampson (2000), which is Leaf-Ancestor Assessment (LA). A possible drawback of applying different metrics of evaluation is that their notions of the degree of correctness can vary from question to question, i.e. a question can reach a high score in the one metric and a rather low in the other. Sampson and Babarczy (2003) have compared the Parseval labelled F-score and the LA score and concluded that there is only a small correlation. However, we will show in the next section that the judgements of the two metrics are highly correlated for our data set of 233 *why*-questions. An explanation for the fact that the two metrics are more similar for our data set than they were for Sampson and Babarczy's (2003) data is that our data set is more uniform because all instances are *why*-questions. The high correlation allows us to employ LA here without running the risk of presenting results that largely diverge from those presented in the previous section.

The calculation of the LA score can best be explained by means of an example. Figure 8.2 shows a syntactic tree of the question *Why are 4300 additional teachers required?*, in which *4300 additional* and *teachers* have been incorrectly analysed as two separate NP's.

Starting from a terminal element, i.e. a leaf in the tree, one moves up in the tree and registers each node label of the desired information level until one reaches the root of the tree. If necessary, squared brackets are inserted in the label sequence to delimite branches with multiple nodes. For *4300 additional*, for example, the category label sequence is *NUM NP S TXTU*. Similarly, a category label sequence can be determined for *4300 additional* in the correct syntactic analysis, which should include brackets because *4300 additional teachers* is a multi-node branch: *NUM [NP S TXTU*. The two label sequences are then compared by applying the minimum edit distance, where deletion and insertion have a penalty of 1, and substitution a penalty of 2. The minimum edit distance for the two label sequences mentioned is 1 (being a deletion of the bracket). The LA score is calculated by subtracting the minimum edit distance from the total number of labels (including brackets) in output and gold standard together, and dividing this again by the total number of labels and brackets. In the example the LA score is (9-1)/9 = 0.89. Combining the scores for all terminal elements indicates the score for the whole sentence.

```
NOFU,TXTU ()
 ├── UTT,S (act,indic,intens,inter,pres,unm)
 │    ├── A,AVP (inter)
 │    │    └── AVHD,ADV (inter) – Why
 │    ├── INTOP,LV (indic,intens,pres) – are
 │    ├── SU,NP ()
 │    │    └── NPHD,NUM (frac,sing) – 4300 additional
 │    ├── CS,NP ()
 │    │    └── NPHD,N (com,plu) – teachers
 │    └── A,CL (-su,intr,pass,pastp,unm,zsub)
 │         └── V,VP (indic,intr,pass,pastp)
 │              └── MVB,LV (indic,motr,pastp) – required
 └── PUNC,PM (inter,qm) – ?
```

Figure 8.2: Example of a syntactic tree: *Why are 4300 additional teachers required?*

Likewise a score can be determined for the whole data set.

A disadvantage of this metric is the fact that errors in nodes high in the tree, dominating many words, have more influence on the scores than errors in lower nodes, dominating fewer words (Sampson et al. 1989, Sampson 2000). The benefit of Leaf-Ancestor Assessment is two-fold. Firstly, we use the minimal edit distance component in the LA metric for (1) analysing the tree structure, and (2) analysing the selection of categories, functions and attributes. Insertions and deletions indicate that there are too few or too many nodes in the tree, denoting incorrect tree structure. For example, there are too many labels for the verb *required* in Figure 8.2 due to the fact that it has incorrectly been parsed as a separate clause. Substitutions involve instances where nodes have been labelled incorrectly. For example, if the attribute ‘passive’ occurs instead of ‘active’, this is a label error in passivity within the attribute type of information. Secondly, the LA scores obtained for individual words or compounds can be used for listing those that fail most often, i.e. those that have the highest proportion of scores lower than 1 (1 being a perfect score). This helps in locating errors as well.

8.3.2 Results

Figure 8.3 shows a comparison between the Parseval labelled F-score and the LA score for our data set, following the example in Sampson and Babarczy (2003). The figure shows the scores for categories in the TOSCA output based on manually verified tagged input (SEMI). The focus is on categories since those are the labels that most other syntactic parsers produce. The correlation between the scores is very high (0.94). Thus, contrary to conclusions in Sampson and Babarczy (2003), both scoring metrics are highly correlated for our data set. The similarity provides us with enough support to use either method, depending on which suits the evaluation purpose best.

The LA scores for the TOSCA parses in SEMI are presented in Table 8.2. The

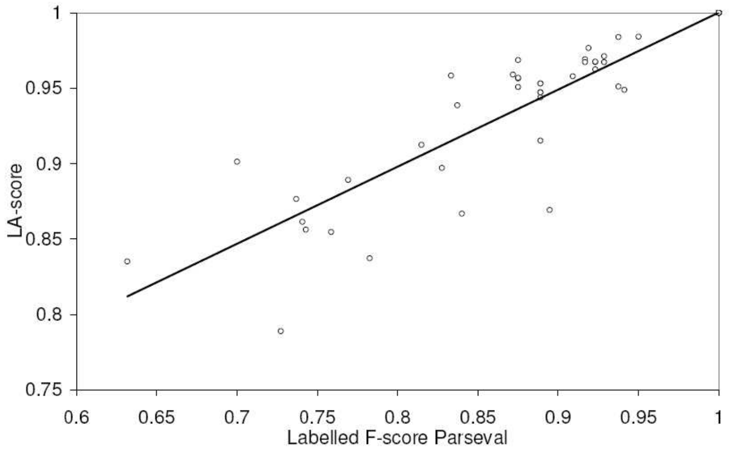

Figure 8.3: Scores for 233 TOSCA parses based on edited input (SEMI), calculated by the two metrics: Labelled F-score following Parseval and LA-score following Leaf-Ancestor Assessment

differences between the scores for categories, functions and attributes are significant (p = 0.000 for all three pairs, following the paired (dependent) t-test). The scores for categories are highest, those for functions lowest. As established in the previous section, more than 80% (188 questions) of the parses are a complete match of the gold standard.

Table 8.2: LA scores for TOSCA output in SEMI

	Categories	Attributes	Functions	Average
LA Score	0.988	0.983	0.976	0.982

Table 8.3 shows a list of words obtaining an LA score lower than 1 (being the perfect score). The first number shows the proportion of occurrences with an erroneous label sequence and the second the average LA scores obtained for all occurrences of the word. The LA scores are the average of the scores obtained for category, function, and attribute(s). We have only listed words that have a frequency of at least five, of which at least a quarter has an imperfect label sequence. This decision prevents inclusion of unique or rare words that have an imperfect analysis: if a word occurs only once in the data set and its label sequence contains an error, 100% of this word fails, which would undesirably position it high in the list.

Table 8.3: Words with an imperfect label sequence.

word	prop.	LA	word	prop.	LA	word	prop.	LA
than	0.60	0.80	*dictionary*	0.40	0.89	*at*	0.30	0.88
chefs	0.60	0.82	*with*	0.38	0.76	*women*	0.29	0.70
for	0.47	0.77	*about*	0.33	0.83	*and*	0.29	0.88
court	0.44	0.80	*warming*	0.33	0.88	*up*	0.25	0.81
supreme	0.43	0.79	*rights*	0.33	0.88	*in*	0.25	0.84
easier	0.40	0.68	*global*	0.33	0.91			

8.3.3 Discussion

There are two indicators of tree structure in the LA metric, being the position of brackets and the number of labels in the label sequences for each terminal element. In 36 questions of the 233 in the data set, there was an error in the placing of brackets. Brackets are only placed when a node has one or more sisters, so an incorrect placement of brackets is a straightforward clue for erroneous tree structure. The other sign of imperfect tree structure is the lack or surplus of node labels in a sequence. This was the case in the same 36 questions plus one other.

An example of an incorrect analysis is that yielded for the question *Why are films planned for release only overseas?*, in which *planned ... overseas* is incorrectly parsed as a postmodifier of the noun *films* (figure 8.4). The word *planned*, for instance, shows that the use of brackets fails and there is a lack of nodes (for categories: *LV VP [CL NP S TXTU* versus the gold standard *LV VP S TXTU*). Both observations help in establishing that the tree structure is erroneous and in locating in what part of the tree the errors occur.

Substitutions of node labels demonstrate incorrect label selection. They especially occur for the attributes and functions selected by the TOSCA parser and to a less extent for categories. In 7 questions, the clause tense was incorrect, for instance by mistaking a progressive construction for a present participle construction. In a few other questions (3), there were problems concerning modality or voice. Errors in the functions ‘subject’, ‘subject complement’, ‘direct object’ and ‘adverbial’ occur in 28 questions. Of these 28 questions, the transitivity of the main clause (*UTT,S* was wrong in 5 questions, in all of which a monotransitive main clause was erroneously parsed as an intransitive one. Since the parser was offered manually checked tags, the transitivity of the verb in the parse must be correct. The problem is that the monotransitive verb is erroneously placed in a subclause, making the subclause monotransitive and the main clause intransitive. This again leads to an erroneous assignment of the function labels ‘subject’ and ‘adverbial’ to elements in the non-existent subclause. In 9 questions, the question word *why* was incorrectly parsed as a subject complement instead of an adverbial. Because of this the word order feature ‘pre-cs’ instead of ‘unmarked’ is selected, meaning the fronting of a subject complement. The remaining 14 questions involving the functions mentioned have too diverse causes to describe them here.

```
NOFU,TXTU ()
├── UTT,S (indic,inter,intr,pass,pres,unm)
│   ├── A,AVP (inter)
│   │   └── AVHD,ADV (inter) – Why
│   ├── INTOP,AUX (indic,pass,pres) – are
│   └── SU,NP ()
│       ├── NPHD,N (com,plu) – films
│       └── NPPO,CL (-su,indic,intr,pass,pastp,unm,zero)
│           ├── V,VP (indic,intr,pass,pastp)
│           │   └── MVB,LV (indic,motr,pastp) – planned
│           ├── A,PP ()
│           │   ├── P,PREP () – for
│           │   └── PC,NP ()
│           │       └── NPHD,N (com,sing) – release
│           └── A,AVP (gen)
│               ├── AVPR,AVP (excl)
│               │   └── AVHD,ADV (excl) – only
│               └── AVHD,ADV (gen) – overseas
└── PUNC,PM (inter,qm) – ?
```

Figure 8.4: Example of TOSCA output in SEMI for the question *Why are the attorneys for the Bush administration present at the hearing?*

The word list in Table 8.3 enables us to locate difficulties in parsing the data we used. Interesting is the large number of prepositions in this list despite the fact that for the greater part, PP-attachment is determined by syntactic markers that we manually inserted prior to the parsing process. The list also shows word groups that occur in the same questions. The words *dictionary*, *easier* and *than*, for instance, are all used in questions posed to a newspaper text about compiling a Spanish equivalent of the Oxford English Dictionary (OED). It appears that though formulated by different native speakers of English, the questions have a similar structure. This is likely to be caused by the design of the elicitation experiment, where participants had access to the news paper texts while formulating questions to them. In questions to other texts, co-occurring words are *court*, *supreme*, *rights* and *women*, and *warming* and *global*. Employing a larger data set with more syntactic and lexical diversity to verify whether the results at the word level are representative for *why*-questions in general is beyond the scope of the present evaluation.

Due to the level of detail in the TOSCA output, it is difficult to compare the results to those obtained by other parsers and to establish a baseline. Often parsers only provide categories in their hierarchical structures, which is also the information level on which TOSCA reaches the highest LA scores. Functions are not commonly included in syntactic analyses due to the fact that they are less obvious to determine. This is confirmed by the lower LA scores for functions that have been obtained by TOSCA. Although a comparison with either other parsers or a baseline cannot be made and not all three levels of information are equally successful, we assume that the LA (0.982) and perfect match scores (0.807) are sufficient to continue the use of the present descriptive model in future versions of the TOSCA parser. Furthermore, previous research has shown that the level of detail of the TOSCA trees is beneficial to the *why*-question answering system (Verberne et al. 2006b), and the presented results encourage the use of the automatic parser in the *why*-QA system.

8.4 Conclusion and further research

In this paper we have presented an intrinsic evaluation of the TOSCA system, which enabled us to pinpoint difficulties in the system and to formulate suggestions for a future version of TOSCA. Moreover, the use of *why*-questions as data facilitate the extrinsic evaluation of TOSCA in the *why*-question answering system.

TOSCA is an interactive parsing system that aims to yield deep linguistic analyses. The output includes detailed syntactic information in the form of categories, functions and attributes. The level of detail and the interdependence between the different types of information in the descriptive model that is being used entails the risk of causing a domino effect in which incorrect categories and/or attributes lead to the erroneous assignment of function labels to constituents. When provided with correct POS tags and post-edited input, however, more than 80% of the first proposed TOSCA analysis is a perfect match of the gold standard. The parses obtain an average LA score of 0.982. We consider the evaluation results sufficient to assume that the level of detail does not really affect the parse accuracy, and is therefore justified in a future version of TOSCA as well.

The modularity of the current TOSCA system is fatal: Tagging errors and missing syntactic markers in automatically obtained input radically decrease the coverage, showing that the parser is not at all robust. Moreover, the Parseval labelled F-scores for those questions that could be parsed were much lower (0.783) than those reached when the tags are corrected and the necessary markers are inserted (0.959). A new version of TOSCA is under construction, in which the level of detail in the parses is maintained, while there is no longer a need to separately provide POS tags for the tokens in the input or insert any syntactic markers.

Since the principle adopted in parsing - yielding minimally the one correct analyses for a given input string - is held onto also with the new implementation of the TOSCA system, the ranking of syntactic parses remains a topic of interest. Future research should be directed at investigating whether and how it would be possible to rank the parses in such a way that the contextually appropriate one is presented as the first one. A possible method to consider is the use of the outcome of the parser evaluation applying the Parseval or LA metric. Each presented parse could then be compared to the gold standard and ranked according to its accuracy score. Subsequently, machine learning algorithms could be employed to find patterns on which general rules for parse ranking can be based. However, such an approach demands a large annotated corpus that is not available at present and should therefore be constructed for this purpose.

References

Aarts, F. and Aarts, J.(1982), *English Syntactic Structures*, Pergamon (Oxford).

Black, E.(1993), Statistically-based computer analysis of English, *in* E. Black, R. Garside and G. Leech (eds), *Statistically-driven computer grammars of English: The IBM / Lancaster approach*, pp. 1–16.

Black, E., Abney, S., Flickenger, S., Gdaniec, C., Grishman, C., Harrison, P., Hindle, D., Ingria, R., F., J., Klavans, J., Liberman, M., Marcus, M., Roukos, S., Santorini, B. and Strzalkowski, T.(1991), Procedure for quantitatively comparing the syntactic coverage of English grammars, *Proceedings of the workshop on Speech and Natural Language*, Leiden, pp. 306–311.

Carroll, J., Briscoe, E. and Sanfilippo, A.(1998), Parser evaluation: a survey and a new proposal, *Proceedings of the International Conference on Language Resources and Evaluation*, Granada, pp. 447–454.

Lin, D.(1995), A dependency-basedmethod for evaluating broadcoverage parsers, *Proceedings of the IJCAI-95*, Montreal, pp. 447–454.

Magerman, D.(1995), Statistical decision-tree models for parsing, *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, Morgan Kaufmann, Cambridge, pp. 276–283.

Oostdijk, N.(1996), Using the TOSCA analysis system to analyse a software manual corpus, *in* R. Sutcliffe, H. Koch and A. McElligott (eds), *Industrial Parsing of Software Manuals*, Rodopi Amsterdam, pp. 179–206.

Quirk, R., Greenbaum, S., Leech, G. and Svartvik, J.(1972), *A Grammar of Contemporary English*, Longman (London).

Sampson, G.(2000), A proposal for improving the measurement of parse accuracy, *International Journal of Corpus Linguistics* pp. 53–68.

Sampson, G. and Babarczy, A.(2003), A test of the leaf-ancestor metric for parse accuracy, *Journal of Natural Language Engineering* pp. 365–380.

Sampson, G., Haigh, R. and Atwell, E.(1989), Natural language analysis by stochastic optimization: a progress report on project APRIL, *Journal of Experimental and Theoretical Artificial Intelligence* pp. 271–287.

van Halteren, H. and Oostdijk, N.(1993), Towards a syntactic database: The TOSCA analysis system, *in* J. Aarts, P. de Haan and N. Oostdijk (eds), *English Language Corpora: design, analysis and exploitation*, Rodopi (Amsterdam), pp. 145–161.

van Rijsbergen, C.(1979), *Information Retrieval*, 2nd edition, Butterworths (London).

Verberne, S., Boves, L., Oostdijk, N. and Coppen, P.(2006a), Data for question answering: the case of why, *Proceedings of the 5th edition of the International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy.

Verberne, S., Boves, L., Oostdijk, N. and Coppen, P.(2006b), Exploring the use of linguistic analysis for why-question answering, *Proceedings of the 16th meeting of Computational Linguistics in the Netherlands (CLIN 2005)*, Amsterdam, pp. 33–48.