Towards Semi-Automatic Analysis of Spontaneous Language for Dutch

Jan Odijk
UiL-OTS
Utrecht University, the Netherlands
j.odijk@uu.nl

Abstract

This paper presents results of an application (*Sasta*) derived from the CLARIN-developed tool *GrETEL* for the automatic assessment of transcripts of spontaneous Dutch language. The techniques described here, if successful, (1) have important societal impact, (2) are interesting from a scientific point of view, and (3) may benefit the CLARIN infrastructure itself since they enable a derivative program that can improve the quality of the annotations of Dutch data in CHAT-format.

1 Introduction

This paper presents results of an application (*Sasta*) derived from the CLARIN-developed tool *GrETEL* for the automatic assessment of transcripts of spontaneous Dutch language. The techniques described here, if successful, (1) have important societal impact, since they enable semi-automatic analysis of spontaneous language in a clinical setting, which is an important ingredient of assessments but requires specialised linguistic expertise and takes a lot of effort; (2) are interesting from a scientific point of view (various phenomena get a linguistically interesting treatment), and (3) may benefit the CLARIN infrastructure itself since they enable a derivative program that can improve the quality of the annotations of Dutch data in CHAT-format (CHILDES data, (MacWhinney, 2000)).

Section 2 introduces methods for the analysis of spontaneous language. Section 3 introduces the CLARIN-developed application GrETEL that Sasta has been derived from. Section 4 briefly describes other work that has been done on automating the analysis of spontaneous language. Section 5 describes the initial experiment that we carried out to assess the potential of the envisaged method. The results were so promising that a small project, called the *SASTA* project, was started up. It is described in section 6. Section 7 describes the most important problems we encountered, and section 8 describes how we addressed a first set of these problems. In section 9 we report on recent results obtained. We end with our conclusions and plans to address the remaining problems (section 10).

2 Analysis of Spontaneous Language

The analysis of spontaneous language is considered an important method for determining the level of language development and for identifying potential language disorders. Crystal et al. (1976) and Crystal et al. (1989) developed the LARSP method for language assessment, remediation and screening.[1] Many researchers developed variants of LARSP for other languages, see e.g. (Ball et al., 2012). Also for the Dutch language various methods have been developed for the analysis of spontaneous language, both for assessment of language development, e.g. GRAMAT (Bol and Kuiken, 1989), TARSP[2], a variant of LARSP for the Dutch language (Schlichting, 2005; Schlichting, 2017), and STAP[3] (van Ierland et al., 2008; Verbeek et al., 2007) as well as for assessment of aphasia, e.g. ASTA[4] (Boxum et al., 2013).

[1] LARSP= Language Assessment, Remediation and Screening Procedure.

[2] Acronym for the Dutch variant of the expansion for LARSP.

[3] Acronym for *Spontane Taal Analyse Tool* 'Spontaneous Language Analysis Tool'.

[4] Acronym for *Analyse voor Spontane Taal bij Afasie* 'Analysis for Spontaneous Language in case of Aphasia'.

Jan Odijk 2021. Towards Semi-Automatic Analysis of Spontaneous Language for Dutch. *Selected papers from the CLARIN Annual Conference 2020*. Linköping Electronic Conference Proceedings 180: 180 165–175.

Though analysis of spontaneous language is important, it requires specialist linguistic knowledge and expertise, is very time consuming and requires full concentration, so that there is a clear need to investigate whether the process can be automated in full or partially.

The whole process involves several steps. We distinguish here four major stages.[5] An assessment procedure starts with a session with the child or a patient to elicit spontaneous speech. This session is recorded. In the second stage, the recording is transcribed. This is a very intense and time-consuming process. Fully or partially automating this stage is highly desirable, and we are investigating it but it is not the focus of this paper. In a third stage the transcript is annotated grammatically, and these annotations are used to make an assessment of the patient's language development or language disorder. Finally, remediation goals and procedures are defined.

All assessment methods define so-called *language measures*. Each language measure defines a particular linguistic phenomenon, e.g. a grammatical construction or a morphological property of a word. Each occurrence of this phenomenon in the spontaneous language transcript is marked by a code, and the number of occurrences of this code determined, and used in a comparison with reference data to assess whether there are any deviations, development disorders or what the nature of the aphasia is. A concrete example from TARSP is the language measure indicated with the code *Hww i*, i.e. auxiliary with infinitive. The number of occurrences of the grammatical construction in which an auxiliary co-occurs with an infinitive as its complement is determined and compared to reference data. In the appendix of (Schlichting, 2005) five examples of this construction occur, as in (1). I added the utterance identifier and bolded the auxiliary and the infinitive:

(1) Examples of *Hww i* in the (Schlichting, 2005) appendix

a. **blijft** die wel **staan** (10)
stays that indeed stand
'Will it continue to stand up'

b. **kan** je **vastmaken** (11)
can one fasten
'one can fasten it'

c. ik **wil** blauwe ogen **tekenen** (13)
I want blue eyes draw
'I want to draw blue eyes'

d. wat **moeten** we met die stukkies **doen** ? (22)
what must we with those pieces-DIM do ?
'What should we do with those little pieces?'

e. weet je wat ik **kan doen** (28)
know you what I can do
'Do you know what I can do'

These sentences must be annotated with the code *Hww i*, and this is done manually.

This paper reports on initial experiments to partially automate the grammatical annotation stage of this process, with the goal to gain efficiency and possibly also to increase the quality of the annotations.

3 GrETEL

GrETEL (Augustinus et al., 2012) is an application to query treebanks. It makes existing manually verified treebanks for Dutch such as *LASSY-Small* for written Dutch (van Noord et al., 2013) and the *Spoken Dutch Corpus* (Oostdijk et al., 2002) available for search. The syntactic structures inside the treebanks are encoded in XML. GrETEL offers XPath to search in these syntactic structures for words, grammatical properties and constructions. In addition, it offers query-by-example facilities.

Version 4 of GrETEL, GrETEL4, (Odijk et al., 2018), enables a researcher to upload a text corpus and associated metadata, and have it automatically parsed by the Alpino parser (Bouma et al., 2001), after

[5] (Crystal et al., 1989) distinguish seven stages. We collapsed some of their stages.

which the resulting parsebank[6] is made available for search. It also offers various ways of analysing the search results, for data and metadata combined.

The text corpus upload functionality also makes it possible to upload a transcript of a spontaneous language session and to analyse it for grammatical properties. We describe an experiment with this in section 5.

4 Related Work

To our knowledge, Bishop (1984) was the first to propose and partially implement an automation of LARSP (for English). Long et al. (1996 2000) developed a different system for English. For French, F-LARSP was automated by Parisse et al. (2012), though it could deal reliably only with inflectional properties and the lower stages of development (stages I–III but not stages IV and V). To our knowledge, no attempts have been made before to automate any of the spontaneous language assessment methods for Dutch. However, there has been work on automating the determination of readability, e.g. with the tools *T-Scan*[7] and *Lint* (Pander Maat and Dekker, 2016; Pander Maat, 2017). Though these are different applications applied to a different domain (prepared written texts) and with different purposes (readability assessment), many of the underlying technologies are shared. For example, T-Scan also uses the Alpino parser. van Noord et al. (2020) developed a Syntactic Profiler of Dutch (SPOD[8]), as part of the treebank query application PaQu (Odijk et al., 2017). SPOD also targets prepared written texts.

5 Schlichting Appendix Test

In order to assess the potential of GrETEL for automating the TARSP analysis, we experimented on the appendix of (Schlichting, 2005). This appendix is intended for illustrating the TARSP analysis and contains a number of example sentences together with their analysis in the form of annotations. We use the analysis as our reference material. For reasons that will become clear below, we call this the *Bronze* reference. The utterances themselves together with their utterance identifiers have been entered in a plain text file in a format supported by GrETEL4.[9] This file has been uploaded into GrETEL4, which results in a parsebank. This parsebank is publicly available in the GrETEL4 application.[10]

An example utterance and analysis is provided in table 1.[11]

Utt	**Level**	**word1**	**word2**	**word3**	**word4**	**ann**	**stages**
10	**Utt**	blijft	die	wel	staan		
10	**gloss**	stays	that	indeed	stand		
10	**Zc**	W	Ond	B		+Inv	III,III
10	**Wg**	Hww i					III
10	**VVW**		AVn				I

Table 1: Example TARSP analysis

The *Utt* column contains the utterance identifier (*10*). The *Level* column contains the label *Utt* for the actual utterance and labels for the levels of analysis: Zc (sentence constructions), *Wg* (word groups), and *VVW* (Connectives, pronouns, and word structure). Next, there as many columns as there are word occurrences (word1, ..., word4), followed by an annotation column for annotations that are not aligned to any specific word occurrence. The final column contains the stages (of language development) that the annotations at that level belong to. The annotations at the *Zc* level are aligned to specific words (*W*=verb,

[6] We call a text corpus in which each sentence has been assigned a syntactic structure automatically a *parsebank*; if the syntactic structures have been manually verified we speak of a *treebank*.

[7] `https://webservices-lst.science.ru.nl/tscan`.

[8] `https://paqu.let.rug.nl:8068/spod`

[9] `https://surfdrive.surf.nl/files/index.php/s/Arsz81uZWbD10z8`.

[10] `http://gretel.hum.uu.nl/gretel-upload/index.php/treebank/show/tarvb2`.

[11] The *gloss* row does not belong to the analysis but has been added here for convenience. The translation of this utterance is 'Does that one really stay standing up'.

Ond=subject, *B*=adverb) except for *+Inv* (=with inversion). The phenomena associated with these annotations belong to developmental stage III. At the *Wg* level the annotation *Hww i* stands for 'auxiliary verb with an infinitival complement'. It is aligned to the auxiliary verb *blijft* but also (implicitly) annotates the infinitive *staan*. Finally, at the *VVW* level, the word *die* has been annotated as a substantively used demonstrative pronoun (*AVn*), typical for stage I.

Though some annotations are aligned to specific word occurrences, the actual usage of such alignments is rather inconsistent. We have disregarded the alignment in the experiment.

The annotated data have been encoded in TSV-format and made available in an Excel file in the data folder.[12] Queries have been written for the TARSP language measures that cover the annotations. These queries yield a list of matches in the GrETEL4 application. The queries themselves as well as URLs which execute these queries directly on the parsebank in GrETEL4 are included in a file that contains a summary of the whole analysis.[13] This file also contains, for each match, the utterance identifier for the utterance in which the match was found. Multiple matches can occur in the same utterance, so the queries yield multisets of utterance identifiers. The Bronze reference has also been specified with a multiset of utterance identifiers for each language measure.

The query used for the code *Hww i*, used as an example in section 2, yields the utterances with utterance identifiers 10, 11, 13, 22 and 28, exactly corresponding with the manual analysis.[14]

We noticed after doing several experiments that GrETEL finds many matches that are (in our view) correct though they do not occur in the Bronze reference. We therefore created a second, improved reference, which we have called the *Silver* reference, which includes the utterance identifiers found by GrETEL that are not in the Bronze reference and had them judged for correctness by one of the clinical linguists that we cooperate with. We suspect that the omission of these annotations in the Bronze reference is partially due to human oversight, and partially due to the fact that these data were never created as reference data but rather as illustrative analyses. Though a comparison with a Silver reference probably yields a higher score than a comparison with a truly complete reference (a *Gold* reference), it is a useful way to get an impression of what kind of performance is attainable. Having a Silver reference enables us to do three comparisons: (1) GrETEL v. Bronze reference, as a measure of quality; (2) GrETEL v. Silver reference, as an improved measure of quality; (3) Bronze v. Silver reference, as a measure of the quality of purely human annotation.

For this experiment, we wrote initial versions of queries to implement the TARSP method, but we only wrote queries for language measures that occur in the Schlichting appendix.

We use *recall*, *precision* and *F1-score* as defined in (2) as performance measures. Here *O* is the multiset of results and *R* is the reference multiset:

(2) Performance measures:

a. Recall: $\frac{|O \cap R|}{|R|}$ (undefined when $|R| = 0$)

b. Precision: $\frac{|O \cap R|}{|O|}$ (undefined when $|O| = 0$)

c. F1-score: $\frac{2 * Recall * Precision}{(Recall + Precision)}$

The results of the experiment have been summarised in table 2.

The figures that we observe here are promising, though it must of course be noted that the experiment has not been carried out on an independent test set. Also note that recall of the automatic system when compared to the silver reference (0.89) is slightly higher than the recall of the human annotation (0.88): inspecting the relevant examples shows that this is caused by the fact that human experts easily overlook instances. However, humans clearly remain superior for precision (0.90 for human annotation, 0.86 for annotation by the system).

[12] `https://surfdrive.surf.nl/files/index.php/s/jJvj16TsDprIKXb`

[13] `https://surfdrive.surf.nl/files/index.php/s/P7lis33HVDgbsKK`.

[14] This link executes this query in GrETEL: `http://shorturl.at/kzEH3`.

Comparison / Measure	R	P	F1
GrETEL v. Bronze	0.88	0.79	0.83
GrETEL v. Silver	0.89	0.86	0.87
Bronze v. Silver	0.88	0.90	0.89

Table 2: Performance of GrETEL versus a human-created Bronze reference, versus an improved reference called Silver, and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1).

6 The SASTA Project

The results described in section 5 were considered promising by ourselves and the Dutch Association of Clinical Linguistics (VKL). For this reason we decided to extend the development, in a project called SASTA (acronym for a Dutch expansion meaning Semi-Automatic Assessment of Spontaneous Language).

In the project we have developed a research prototype application called *Sasta* aimed at clinical linguists that takes as input (1) a transcript to be analysed; and (2) an assessment method to be applied. The application yields as output (1) a standard profiling form in accordance with the assessment method, plus an assessment of the language development stage or the language disorder of the patient; (2) the transcript enriched with annotations. The automatically annotated transcript can be manually adapted and then offered to *Sasta* again for generating a revised profiling form. We support three different assessment methods (TARSP, STAP and ASTA). Each method is defined as a set of queries, special modules that are needed, measures to deal with deviating input, etc. associated to language measures of the method.

In order to develop *Sasta* we have developed *Sastadev*, a piece of software intended for developers that enables input of multiple reference data in multiple formats and compares the output of *Sasta* with the references and provides a detailed analysis of the differences. Many data provided by VKL members and other clinical linguists have been used for developing the system.

Sasta and *Sastadev* reuse components of GrETEL (the Alpino parser, the upload functionality, and the query functionality) but apply them differently: GreTEL is optimally suited to apply a single query to a large treebank, while *Sasta* and *Sastadev* are more suited to apply multiple queries to a small treebank.

Automating TARSP requires formalising certain aspects of the methods. For example, Schlichting herself uses codes in her examples that are not defined in the definition of the method, though they resemble them.[15] Data that we received from clinical linguists sometimes use yet other variants of the codes. Multiple annotations on a single word are separated by a hyphen or space though hyphens and spaces also occur inside codes (e.g. in *aan-uitloop* and *hww i*). It also appears that the coding scheme does not use fixed codes but presupposes a productive syntax. However, the coding scheme has not been formalised, and uses natural language words, which gives rise to all the horrors of natural language.[16] We have formalised the annotations while at the same time allowing as much flexibility as possible to accommodate actual practice.

7 Problems to Be Addressed

There are many problems that the data and the technology pose and that have to be addressed.

First, the transcripts of the spontaneous language sessions contain a large amount of deviations of normal language use. These are partially due to annotation conventions, and partially due to the fact that the children who are still learning the language and patients with aphasia make imperfect utterances.

[15]For example *+inv* instead of *inv*; *v.u.soc.divers* instead of *v.u. sociale uitdrukkingen*; *neg* instead of *xneg*, etc. And in the form provided, sometimes yet other codes are used (e.g. *V.U.Soc.Ster* v. *v.u. sociale uitdrukkingen: stereotiepe uitdrukkingen*. Most of them are easily interpretable by humans but not by software.

[16]Successful communication is seriously hampered by natural language, even in as simple a domain as words or terms: natural language words have associations, have a (common sense) meaning, are often ambiguous, are specific to one language, and have variations (abbreviations, acronyms etc.). These properties make successful communication difficult if not impossible, surely between humans and machines but often also between humans. It is much better practice to use arbitrary labels that at best resemble existing words for mnenomical reasons but that are no natural language words.

Conventions for annotating the data had to be made more formal and more detailed. For example, TARSP has the convention that the actual utterance can be accompanied by additional remarks by the annotator between round brackets. However, such round brackets contain two different types of annotations: (1) indication for non-existing words: which word was intended by the patient, according to the annotator; (2) other remarks by the annotator. We want to make optimal use of these annotations, but then these two different uses must be formally distinguished. We therefore require annotations based on the CHAT-format, which formally encodes these two different cases differently.

All kinds of deviations occur in the transcripts. Here is a list of the most common deviations:

- Often, a string is a non-existing word because the transcript also describes how the word was pronounced, e.g. *mouwe* instead of mouwen 'sleeves' with the *n* unpronounced; *isse* instead of *is een* 'is a', *zie-ken-huis* with hyphens to indicate the separated pronunciation of the syllables of the word *ziekenhuis* 'hospital'.

- overregularisation of word forms (e.g. *gevald* instead of *gevallen* 'fallen'), and even misspellings of such overregularisations (*gevalt*).

- wrong inflected forms, e.g. *gekeekt* instead of *gekeken* 'watched').

- filled pauses.

- dialectical or sociolectical form variants, e.g. *-ie*-diminutives. (*boekie*) instead of *(t)je*-diminutives (*boekje* 'booklet').

- repetitions of (sequences of) word occurrences.

- partial repetitions of repeated words.

- false starts.

- other often-occurring grammatical errors, e.g use of the wrong article, or of the wrong auxiliary for perfect tenses, agreement errors, etc.

For some of these, we are quite confident that we can address them in a sufficiently reliable manner to improve the analysis, and we have made some initial steps towards this. For others, however, we are less confident but we will nevertheless investigate how far we can get.

Second, the Alpino parser has limitations. It cannot analyse all compounds as compounds, it provides insufficient information on verbless utterances, it provides insufficient information on verb-first sentences, it sometimes parses an utterance incorrectly, it sometimes analyses an utterance in a way that differs from the reference (but is not incorrect). Alpino does not consider the context, can do very little when semantic restrictions apply, and cannot deal with intonation .

Third, certain items require queries that cannot be expressed in XPath or only with great difficulty, e.g., the TARSP item *6+* which requires 6 or more constituents in a clause, or the STAP query for adverbs other than locative and temporal adverbs (this query takes up 315 lines in XPath!).

8 Towards Solutions

Many of the problems identified in section 7 can be addressed and several have already been addressed.

For example, by writing the right queries we can analyse certain adverbs inside phrases as if they occur at a sentential level. For queries that cannot be easily formulated in XPath we enable functions in a full programming language (we use Python). In addition, we allow macros inside XPath queries to make the queries shorter and easier to read and to facilitate reuse. For example, the definition of 'auxiliary verb' in Tarsp requires a long enumeration of lemmas, and this exact same enumeration must be used in two different queries (*Hww i* and *HwwZ*). With macros the enumeration has to be stated only once.

We developed new modules for normalising orthography, for analysing compounds, for dealing with regional spoken language diminutives ending in *-ie(s)*,[17] for overgeneralised inflectional forms of verbs (even misspelled ones), and for automatically detecting filled pauses and repetitions. We use these to adapt each utterance that Alpino cannot deal with to a variant of this utterance that Alpino can deal with. Some examples have been given in table 3.

Original utterance	**Corrected utterance**	**Gloss**
mama mouwe hoog	mama mouwe**n** hoog	mum sleeves high
niet goed uitgekijken	niet goed uitgek**e**ken	not well looked-out
die stukkies	die stuk**j**es	those pieces-DIM
zie-ken-huis	ziekenhuis	hospital

Table 3: Some examples of automatic corrections to improve the performance of Alpino.

We are using data provided by the VKL and by several clinical linguists, all example sentences of (Schlichting, 2005) and Dutch CHILDES data during development.

We use these modules to generate 'corrected variants' of deviant utterances, so that Alpino can parse the utterance correctly. The system annotates each utterance for the errors encountered and the corrections applied, so that also an error analysis results. After parsing the corrected utterance the system replaces the corrected words by the original words on the basis of the metadata.[18]

We have also developed a module to automatically detect filled pauses and repetitions, and are experimenting with a module for automatically detecting false starts.

9 Recent results

Schlichting	**%**		O v B			O v S			B v S		
Eval Meth	**Corr**	**Exts**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
Sastadev	No	No	86.5	80.8	83.6	88.4	89.3	88.8	89.8	97.0	93.3
Sastadev	Yes	No	88.5	81.2	84.7						
Sastadev	No	Yes	88.0	70.1	78.0	89.0	77.1	82.6	86.8	94.4	90.4
Sastadev	Yes	Yes	91.2	70.8	79.7						

Table 4: Performance of Sastadev (version of early 2020) for the Schlichting Appendix (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1). Results are given for the original version of TARSP, i.e. the version also used in the initial experiment described in section 5 (*Exts=No, Corr=No*), for the original version of TARSP with corrections (Exts=No, Corr=Yes), and for an extended version of TARSP without (*Exts=Yes, Corr=No*) and with (*Exts =Yes, Corr=Yes*)corrections.

Table 4 shows the performance of Sastadev in the version of early 2020 for the Schlichting Appendix (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1).[19] Results are given for the original version of TARSP (*Exts=No, Corr=No*), for the original version of TARSP with corrections (Exts=No, Corr=Yes), and for an extended version of TARSP without (*Exts=Yes,*

[17]This entails more than just replacing an *i* by a *j*, e.g. *bekkie* corresponds to *bekje* 'beak' , *bekie* to *beekje* 'brook', *cluppie* to *clubje* 'club', etc.

[18]Alpino actually provides some facilities for this, by so-called bracketed input, but we decided to use our own implementation in SASTA.

[19]The scores of the automated comparison differ from the manual comparison (as in table 2) because some codes were wrongly counted in the manual comparison, e.g. the word *stukkies* 'small pieces' was wrongly analysed as a singular compound (*stuk* 'broken' + *kies* 'tooth') instead of as a plural diminutive, and this was counted as single error; but it should have been counted as three errors.

Corr=No) and with (*Exts =Yes, Corr=Yes*) corrections. The corrections applied are certain orthographic normalisations (addition of *n* after a word ending in *e*, separation of incorrectly concatenated words, dehyphenation), normalisation of regional diminutives, normalisation of overgeneralisations for verbs, and a compound identification module). These corrections are currently only applied for words that are not contained in a Dutch lexicon or in a name list, and at most one variant is considered.

We note a significant drop in precision for the version with the extensions (from 80.8 to 70.1 for the Bronze reference, and from 89.3 to 77.1 for the Silver reference). This is caused by the fact that the extensions, which involve improved versions of existing queries as well as new queries that were not defined before, cause SASTA to identify even more hits that were not marked in the manual annotations, and, for the new queries, even had no occurrence at all in the reference. For this reason we created a new Silver reference for the Schlichting appendix. Table 5 contains the scores for the current version of SASTA (early 2021). Compared to this new silver reference, SASTA scores higher than 90% for recall, precision and f1-score for the Schlichting Appendix. The corrections improve the recall by more than 2 percent points also here, and improve precision marginally.[20]

Schlichting	**%**	O v B			O v S		
Eval Meth	**Corr**	**R**	**P**	**F1**	**R**	**P**	**F1**
Sastadev	No	90.1	72.4	80.3	92.0	91.4	91.7
Sastadev	Yes	93.1	73.1	81.9	94.4	91.6	93.0

Table 5: Performance of Sastadev (version of January 29, 2021) for the Schlichting Appendix (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), in terms of recall (R), precision (P) and F1-score (F1). Results are given for the most recent extended version of TARSP without corrections (*Corr=No*), and for the most recent extended version of TARSP with corrections (Corr=Yes). Note that the Silver reference used here differs from the one used in table 4.

We are still developing our implementation of TARSP and the SASTA modules. In the current implementation the corrections that we experimented with for the Schlichting Appendix have not been integrated yet. The results of the most recent system[21] are given in table 6 (for the TARSP data), in table 7 (for the STAP data), and in table 8 (for the ASTA data).

We have worked with too little data to be able to keep a subset of the data separate, so here we can only report on results on data that have been used during the development of the system. However, we still do have independent data, and hope to report on results on these data in the near future when they have been converted to a format usable by Sastadev. We observe that recall for TARSP compared to the Bronze reference is of reasonable quality, but precision is rather low. However, compared to the Silver reference, precision increases dramatically (for Sample_08 more than 23 percent points!) and recall is also higher when compared to the Silver reference. We observe also here that recall of Sastadev is sometimes higher than recall in the purely human annotation (e.g. in samples 4, 7, 8, 9, 10) though also here precision by human annotators remains superior to Sastadev.

For STAP, we observe that recall and precision are already pretty good when compared to the Bronze reference, and also here they both increase when compared to the Silver reference.[22]

For ASTA, we do not have Silver reference data for all samples yet. However, for the Silver reference data that are available, we see again that precision increases dramatically, and also recall increases. The ASTA scores overall are lower than the scores for TARSP and STAP, and this is very likely caused by the fact that the ASTA data contain no annotations for repetitions, false starts, filled pauses or for incorrect words at all, though such phenomena are mostly (though not always) explicitly marked in the TARSP and STAP data. The queries for detecting filled pauses, repetitions, false starts, and incomplete sentences are very complicated and currently score relatively low. This also affects the results for certain other queries (e.g. for nouns and main verbs), since words must be annotated for part of speech differently

[20] We did not make the Bronze v. Silver comparison for these data yet.

[21] Measurement done on 2021-01-22.

[22] For STAP we never received the reference annotations for sample_01, so we report only on the results for 9 samples.

TARSP	O v. B			O v. S			B v. S		
Sample	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
Sample_01	85.1	72.2	78.1	87.2	86.1	86.6	85.9	100.0	92.4
Sample_02	87.1	64.3	74.0	87.9	69.0	77.3	92.9	98.9	95.8
Sample_03	77.3	63.6	69.8	81.4	81.4	81.4	82.2	100.0	90.2
Sample_04	93.2	81.2	86.8	93.9	90.0	91.9	90.8	100.0	95.2
Sample_05	83.2	69.1	75.5	85.6	83.0	84.3	85.0	99.3	91.6
Sample_06	75.0	56.8	64.7	79.8	75.0	77.3	79.8	99.0	88.4
Sample_07	86.2	66.7	75.2	88.7	80.0	84.1	82.3	96.0	88.6
Sample_08	77.7	63.2	69.7	82.7	86.5	84.6	77.7	100.0	87.4
Sample_09	89.4	69.0	77.9	91.4	87.0	89.1	80.6	99.3	89.0
Sample_10	80.3	69.1	74.3	84.1	89.7	86.8	79.5	98.6	88.1

Table 6: Results of Sastadev for the TARSP data (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1).

STAP	O v. B			O v. S			B v. S		
Sample	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
Sample_02	78.6	79.0	78.8	81.1	92.2	86.3	86.8	98.2	92.2
Sample_03	91.7	85.7	88.6	92.0	89.8	90.9	92.0	96.1	94.0
Sample_04	92.5	93.4	93.0	92.7	95.3	94.0	97.7	99.5	98.6
Sample_05	92.6	85.5	88.9	93.2	92.8	93.0	92.3	99.5	95.8
Sample_06	92.3	88.5	90.4	93.0	97.9	95.4	91.0	100.0	95.3
Sample_07	91.6	92.4	92.0	91.9	96.0	93.9	95.3	98.7	97.0
Sample_08	94.4	79.7	86.4	95.0	90.5	92.7	87.8	99.0	93.0
Sample_09	95.6	84.5	89.7	96.1	95.3	95.7	88.3	99.1	93.4
Sample_10	91.5	80.8	85.8	92.5	92.5	92.5	88.4	100.0	93.8

Table 7: Results of Sastadev for the STAP data (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1).

when they are in a repetition or a false start. For example in a sequence *beits afbeits* 'stain stain-off' *beits* is analysed as a noun though it should have been analysed as a verb since it is a false start of the following word *afbeits*, which is a verb.

10 Concluding Remarks and Future Work

We have presented *Sasta*, an application to analyse transcripts of spontaneous language. Though the *Sasta* application applies only to Dutch, the techniques described here can be applied to any language provided there is a parser for that language and a query system for querying the syntactic structures resulting from the parser. We observe that SASTA scores pretty well on the grammatical analysis of transcripts of spontaneous language sessions. We also found that corrections of deviant language not only improve the deviant parts but also the overall analysis. SASTA also often finds more examples for a grammatical phenomenon than human annotators (who often overlook instances), but the human annotators remain superior in precision. Whether the quality of the grammatical analysis is good enough to make the whole process more efficient remains to be seen. With the VKL we will carry out experiments (starting in January 2021) in which *Sasta* will actually be used in the clinical setting so that we can assess this and optimally integrate *Sasta* into the normal workflow procedures of the hospitals and clinics. In addition, we have secured funding for a small successor project (SASTA+) in which we will investigate more advanced methods for the detection and correction of deviations, including cases in which all

ASTA	O v. B			O v. S			B v. S		
Sample	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
Sample_01	77.0	78.3	77.6						
Sample_02	77.5	78.0	77.7	79.5	85.3	82.3	90.5	96.6	93.5
Sample_03	75.5	78.7	77.1						
Sample_04	79.3	83.4	81.3	81.7	91.0	86.1	92.3	97.7	95.0
Sample_05	73.8	76.4	75.1	78.2	91.8	84.4	86.9	98.5	92.3
Sample_06	95.6	83.8	89.3						
Sample_07	82.1	79.2	80.6						
Sample_08	89.4	78.5	83.6						
Sample_09	89.2	84.3	86.7						
Sample_10	78.8	82.3	80.5						

Table 8: Results of Sastadev for the ASTA data (O) versus a human-created Bronze reference (B), versus an improved reference called Silver (S), and of the Bronze reference versus the Silver reference in terms of recall (R), precision (P) and F1-score (F1). Silver references are currently available only for samples Sample_02, Sample_04 and Sample_05.

words in the utterance are correct and cases where multiple variants should be considered. The automatic corrections developed here can also be used to improve existing CHILDES CHAT annotation files, and we will create a side result of this work, a program to improve and enrich existing CHAT files.

Acknowledgements

I would like to thank the VKL working group members for contributing the data and providing us with their knowledge and expertise on the assessment methods, various other linguists who provided us with data (Mieke Stap, Wim Tops, Jacqueline van Kampen, Liesbeth Schlichting, Eliska Heukels) and my colleagues Jelte van Boheemen and Sjoerd Eilander. This research was funded by CLARIAH-PLUS (an NWO project, Grant 184.034.023), the Dutch Association for Clinical Linguistics (VKL) and the Dutch Language Technology Foundation (Stichting Taaltechnologie).

References

[Augustinus et al.2012] Liesbeth Augustinus, Vincent Vandeghinste, and Frank Van Eynde. 2012. Example-based treebank querying. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, May. European Language Resources Association (ELRA).

[Ball et al.2012] Martin J. Ball, David Crystal, and Paul Fletcher, editors. 2012. *Assessing Grammar: The Languages of LARSP*. Number 7 in Communication Disorders across Languages. Multilingual Matters, Bristol.

[Bishop1984] D. V. M. Bishop. 1984. Automated LARSP: Computer-assisted grammatical analysis. *British Journal of Disorders of Communication*, 19(1):78–87.

[Bol and Kuiken1989] Gerard Bol and Folkert Kuiken. 1989. *GRAMAT: Methode voor het diagnosticeren en kwalificeren van taalontwikkelingsstoornissen*. Berkhout, Nijmegen.

[Bouma et al.2001] Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2001. Alpino: Wide-coverage computational analysis of Dutch. *Language and Computers*, 37(1):45–59.

[Boxum et al.2013] Elsbeth Boxum, Fennetta van der Scheer, and Mariëlle Zwaga. 2013. *Analyse voor Spontane Taal bij Afasie. Standaard in samenwerking met de VKL*. VKL, October. `https://klinischelinguistiek.nl/uploads/201307asta4eversie.pdf`.

[Crystal et al.1976] D. Crystal, P. Fletcher, and M. Garman. 1976. *The grammatical analysis of language disability*. Edward Arnold, London.

[Crystal et al.1989] D. Crystal, P. Fletcher, and M. Garman. 1989. *The grammatical analysis of language disability*. Cole and Whurr, London, 2nd edition.

[Long et al.1996 2000] S.H. Long, M.E. Fey, and R.W. Channell. 1996–2000. Computerized profiling, versions 9.0.3-9.2.7 (ms-dos) [computer program]. Software, Department of Communication Sciences, Case Western Reserve University, Cleveland, OH.

[MacWhinney2000] Brian MacWhinney. 2000. *The CHILDES Project: Tools for Analyzing Talk*. Lawrence Erlbaum Associates, Mahwah, NJ, 3 edition.

[Odijk et al.2017] Jan Odijk, Gertjan van Noord, Peter Kleiweg, and Erik Tjong Kim Sang. 2017. The parse and query (PaQu) application. In Jan Odijk and Arjan van Hessen, editors, *CLARIN in the Low Countries*, chapter 23, pages 281–297. Ubiquity, London, UK. DOI: `http://dx.doi.org/10.5334/bbi.23`. License: CC-BY 4.0.

[Odijk et al.2018] Jan Odijk, Martijn van der Klis, and Sheean Spoel. 2018. Extensions to the GrETEL treebank query application. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories (TLT16)*, pages 46–55, Prague, Czech Republic, January 23-24. `http://aclweb.org/anthology/W/W17/W17-7608.pdf`.

[Oostdijk et al.2002] N. Oostdijk, W. Goedertier, F. Van Eynde, L. Boves, J.P. Martens, M. Moortgat, and H. Baayen. 2002. Experiences from the Spoken Dutch Corpus project. In M. González Rodriguez and C. Paz Suárez Araujo, editors, *Proceedings of the third International Conference on Language Resources and Evaluation (LREC-2002)*, pages 340–347. ELRA, Las Palmas.

[Pander Maat and Dekker2016] Henk Pander Maat and Nick Dekker. 2016. Tekstgenres analyseren op lexicale complexiteit met TScan. *Tijdschrift voor Taalbeheersing*, 38(3):263–304.

[Pander Maat2017] Henk Pander Maat. 2017. Zinslengte en zinscomplexiteit. *Tijdschrift voor Taalbeheersing*, 39(3):297–328.

[Parisse et al.2012] Christophe Parisse, Christelle Maillart, and Jodi Tommerdahl. 2012. F-LARSP: A computerized tool for measuring morphosyntactic abilities in French. In Martin J. Ball, David Crystal, and Paul Fletcher, editors, *Assessing Grammar: The Languages of LARSP*, number 7 in Communication Disorders across Languages, chapter 13.

[Schlichting2005] Liesbeth Schlichting. 2005. *TARSP: Taal Analyse Remediëring en Screening Procedure. Taalontwikkelingsschaal van Nederlandse kinderen van 1-4 jaar*. Pearson, Amsterdam, 7th edition.

[Schlichting2017] Liesbeth Schlichting. 2017. *TARSP: Taalontwikkelingsschaal van Nederlandse kinderen van 1-4 jaar met aanvullende structuren tot 6 jaar*. Pearson, Amsterdam, 8th edition.

[van Ierland et al.2008] Margreet van Ierland, Jeannette Verbeek, and Leen van den Dungen. 2008. *Spontane Taal Analyse Protocol. Handleiding van het STAP-instrument*. UvA, Amsterdam.

[van Noord et al.2013] Gertjan van Noord, Gosse Bouma, Frank Van Eynde, Daniël de Kok, Jelmer van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2013. Large scale syntactic annotation of written Dutch: Lassy. In Peter Spyns and Jan Odijk, editors, *Essential Speech and Language Technology for Dutch*, Theory and Applications of Natural Language Processing, pages 147–164. Springer Berlin Heidelberg.

[van Noord et al.2020] Gertjan van Noord, Jack Hoeksema, Peter Kleiweg, and Gosse Bouma. 2020. SPOD: Syntactic profiler of Dutch. *Computational Linguistics in the Netherlands Journal*, 10:129–145.

[Verbeek et al.2007] Jeannette Verbeek, Leen van den Dungen, and Anne Baker. 2007. *Spontane Taal Analyse Protocol. Verantwoording van het STAP-instrument, ontwikkeld door Margreet van Ierland*. UvA.