The IPP effect in Afrikaans: a corpus analysis

Liesbeth Augustinus[1], *Peter Dirix*[1,2]

(1) Centre for Computational Linguistics, University of Leuven
(2) Nuance Communications, Inc.

{liesbeth,peter}@ccl.kuleuven.be

ABSTRACT

Compared to well-resourced languages such as English and Dutch, NLP tools for linguistic analysis in Afrikaans are still not abundant. In order to facilitate corpus-based linguistic research for Afrikaans, we are creating a treebank based on the *Taalkommissie* corpus. We adapted a tokenizer and a shallow parser, while using a TnT tagger to do part-of-speech annotation. A first linguistic phenomenon we are investigating is the occurrence of *infinitivus pro participio* (IPP) in Afrikaans. IPP refers to constructions with a perfect auxiliary, in which an infinitive appears instead of the expected past participle. The phenomenon has been studied extensively in Dutch and German, but studies on Afrikaans IPP triggers are sparse. In contrast to the former two languages, it is often mentioned in the literature that in Afrikaans, IPP occurs optionally. We want to check this statement doing a corpus analysis.

KEYWORDS: Afrikaans, tokenizer, parser, chunker, corpus search tool, IPP.

1 Introduction

Afrikaans is a West Germanic language spoken as a first language by about 7 million people in South Africa and Namibia and by many millions more as a second language. It can be considered a daughter language of Dutch, as it originates in 17th-century Dutch dialects, brought to southern Africa by settlers from the Netherlands. Although there are some influences from Malay, Portuguese, Bantu, and Khoisan languages, Dutch and Afrikaans are still more or less mutually comprehensible. One of the main features of Afrikaans is a simplification of Dutch morphology, e.g. dropping the nominal gender distinction and only keeping two verb forms for all but the most common verbs (present/infinitive and past participle).

In recent years, several NLP tools were created for Afrikaans, cf. Grover et al. (2011) for an overview of the available tools. Compared to well-resourced languages such as English and Dutch, however, it seems that the tools which are available for Afrikaans are less well-performing.

The purpose of our research is twofold. As a starting point, we describe the NLP tools that were used to process and query the data, as well as the first step towards the creation of a treebank based on an Afrikaans text corpus (the *Taalkommissie* corpus[1]) (cf. section 2).

In the second part of this paper we investigate whether and how the tools and resources that are currently available can be used as a means for descriptive linguistics. As a case study, we will look for the occurrence of *infinitivus pro participio*, a.k.a. the IPP effect, in the *Taalkommissie* corpus. In this linguistic study we compare the IPP phenomenon as it is described in the literature (cf. section 3) to its occurrence in the data (cf. section 4 and 5).

Besides improving the performance of the existing annotation tools, we intend to include the parsed corpus into a user-friendly query engine in order to facilitate corpus-based linguistic research for Afrikaans (cf. section 6).

2 Tools

In order to investigate the linguistic case study described in sections 3 to 5, we automatically annotated the *Taalkommissie* corpus. This section describes the tools used to annotate and query the corpus. We adapted a tokenizer and a shallow parser, while using a TnT tagger (Brants, 2000) trained on Afrikaans to do part-of-speech (PoS) annotation. We furthermore added a search engine to facilitate corpus exploitation.

2.1 Tokenizer

The Dutch tokenizer (Dirix et al., 2005) used in the METIS-II project is rule-based, using regular expressions which model the finite-state characteristics of tokenization and gives only one tokenization per sentence, so the output does not contain any ambiguities. The tokenizer basically splits on white space and detaches punctuation marks from the adjacent words. We adapted the Dutch rules to Afrikaans in order to deal with abbrevations that include a period and the ones to deal with words containing apostrophes (e.g. the indefinite article *'n*).

[1] Taalkommissie van die Suid-Afrikaanse Akademie vir Wetenskap en Kuns (2011).

2.2 Tagger and tag set

We used the TnT Tagger (Brants, 2000), a Hidden Markov Model based *n*-gram tagger, which was trained by CTexT[2] to tag the corpus (further referred to as the *CTexT tagger*). The tag set consists of 139 different tags, based mainly on morphosyntactic features (Pilon, 2005).

In the case of verbs, which is the most relevant PoS for our research (cf. sections 3 and 5), a distinction is made between transitive and intransitive verbs, between separable and inseparable verbs, and also between main verbs, modal verbs, temporal auxiliaries and passivizing auxiliaries. Marked forms (*ge*-marking or simple past in the case of a few auxiliaries) and unmarked forms are also distinguished. Altogether, there are 17 verbal tags, as shown in Table 1.

Tag	Value
VTHOG	inseparable transitive main verb, unmarked
VVHOG	inseparable transitive main verb, marked
VTHOO	inseparable intransitive main verb, unmarked
VVHOO	inseparable intransitive main verb, marked
VTHOV	inseparable intransitive main verb requiring preposition, unmarked
VVHOK	inseparable intransitive main verb requiring preposition, marked
VTHOK	copula, unmarked
VVHOK	copula, marked
VTHSG	separable transitive main verb, unmarked, marked
VTHSO	separable intransitive main verb, unmarked
VTUOM	modal auxiliary, present
VVUOM	modal auxiliary, past
VTUOA	aspectual auxiliary, present
VVUOA	aspectual auxiliary, past
VTUOP	passive auxiliary, present
VVUOA	passive auxiliary, past
VUOT	temporal auxiliary

Table 1: Verbal tags in the CTexT tagger.

The author of the tagger claims an accuracy of 85.87% on a small data set, which is rather low compared to state-of-the-art PoS taggers for well-resourced languages.[3] Although there are some other taggers trained for Afrikaans, they did not seem to meet our research goals. For example, Schlünz (2010) reports an accuracy of 94.64%, but with a tag set reduced to only 17 different tags. The TiMBL-based tagger for Afrikaans (Puttkammer, 2006) is not usable for our purpose, because it mainly identifies different categories of named entities instead of the regular PoS tags.

2.3 Parser

We aim to create a treebank for Afrikaans. As a starting point for syntactic annotations, we used ShaRPa, a shallow rule-based parser (Vandeghinste, 2008) coming with grammars for English and Dutch. In order to parse the *Taalkommissie* corpus, we created an Afrikaans grammar. The different steps can either be defined as context-free grammars, using the PoS tags as preterminals or as Perl subroutines, defined in a Perl module. Note that the grammars are not automatically processed in a recursive mode. The module allows the application of rules which cannot be formulated in the context-free grammar formalism. An

[2]Centre for Text Technology, North-West University, Potchefstroom, South Africa.

[3]The low accuracy is probably due to the fact that the tagger is trained on only 20,000 tokens.

option file defines the application order of the different grammars and subroutines. Both grammars and subroutines can be applied more than once.

Since this is a shallow parser, there is not much depth in the resulting parse tree. It uses the tagged corpus as input, and returns the parsed structure, with marked NPs, PPs, verb groups (VG), and some APs and VPs. The head of a phrase is also marked (`/M`). Each phrase is presented on one line. Each line is divided into three columns: the phrase tokens, the phrase name (assigned by ShaRPa), and the phrase structure, representing the parse building history (containing the PoS tags assigned by the tagger).

An example parse for the sentence *Dis haar handpalms wat begin sweet het, besef sy.* (It is the palms of her hands that had started to sweat, she noticed.):

```
<s>
dis              NP   NP[NSE0[NSE]]
haar handpalms   NP   NP[P00B[PDVEB] NSE0[NSE]/M]
wat              PB   PB
begin sweet het  VG   VG[VP[VTH00] VP[VVH00] VU0T/M]
,                ZM   ZM
besef            NP   NP[NA]
sy               P00B P00B[PDHEB]
.                ZE   ZE
</s>
```

Note that *dis*, a shortened form of *dit is* (it is), is mistagged as the far more infrequent homograph noun (a formal word for 'table') and that *besef*, which can be both a verb (to notice) and a noun (notion), is mistagged as noun.

The verb groups, however, do not give more information than the sequence of tags. As our shallow parser currently does not identify discontinuous verb groups, we will need to introduce a full parse in order to be able to use this information. The quality of the tagging also influences the quality of the parse, so we need to improve the tagger results in order to achieve better results.

2.4 Corpus search tool

In order to look for linguistic constructions in the *Taalkommissie* corpus, we have created a corpus search tool.
The preprocessing consisted of tokenizing and tagging the corpus with the tokenizer and PoS tagger described in section 2.1 and 2.2 respectively. Next, we assigned a unique identifier to each sentence. Then we stored the complete corpus into a PostgreSQL database.[4] For each sentence, we included the following information in the database:

`ID | sentence | PoS string | token-PoS string`

Since the *Taalkommissie* corpus is rather large, we used the (built-in) B-tree indexing aiming to speed up corpus search.

In order to facilitate querying the corpus, we added a search interface on top of the database. The interface is a combination of PHP scripts and HTML, resulting in a web-based search

[4] `http://www.postgresql.org`

tool which allows users to query the corpus without any local installation of corpora and/or software.

As input, the user provides a query which could be a string of tokens, e.g. `het kom kuier` (lit. 'have come visit'), a string of PoS tags, e.g. `[VUOT] [VTUOA] [VTHOG]` (base form of temporal auxiliary, aspectual verb, main verb), or a combination of both tokens and tags, e.g. `het[VUOT] kom[VTUOA] kuier[VTHOG]`. Note that PoS tags should be put between square brackets. It is furthermore possible to use a wildcard for the PoS tags. For example, if one wants to look for any verb form, `[V*]` can be used; if one want to differentiate between base forms and inflected verb forms, `[VT*]` and `[VV*]` can be used respectively.

Furthermore, there is an option to include some context before and after the matching sentences. This might be useful to disambiguate homonyms in the case of short sentences, or if one is interested in discourse phenomena.

After querying the corpus, the results are presented to the user (see screenshot in Figure 1). At the top of the page, the search instruction is repeated. Below the query, a list of matching sentences is displayed. The constructions matching the query are highlighted in each sentence. It is also possible to view/save the results as plain text format (with and without PoS tags).

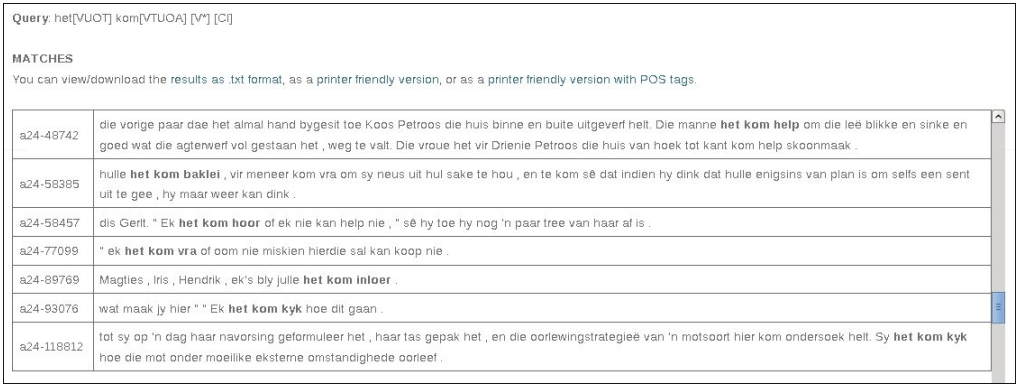

Figure 1: Corpus search tool interface

At the bottom of the page, a grid with the corpus results is printed. It indicates how many hits in how many matching sentences were found. Furthermore, the ratio (matching sentences/sentences in the corpus) is given.

At the moment, it is not possible to query the corpus partially. It might be interesting to look into specific parts of the corpus (e.g. newspaper texts only), but unfortunately the corpus lay-out did not allow us to divide the corpus along those lines.

3 Infinitivus pro participio

3.1 IPP in double infinitive constructions

Infinitivus pro participio (IPP) or *Ersatzinfinitiv* is a linguistic phenomenon occurring in a subset of the West Germanic languages, such as Dutch, German, and Afrikaans. IPP refers to constructions with a perfect auxiliary, in which an infinitive appears instead of the expected past participle. In Afrikaans, one expects the temporal auxiliary for the perfect tense to select

a past participle, marked in various ways, most generally by a prefix *ge-* and sometimes an ending (usually either *-d/-t* or *-en*), cf. *gebly* in example (1a).[5] However, when a verb occurring in the perfect tense selects another verb, it commonly occurs as an infinitive, cf. *bly* in example (1b), instead of the expected past participle, as illustrated in example (1c).[6]

(1) (a) *Hy het stil* **gebly**.
he have:PRES silent stay:PP
'He remained silent.'

(b) *Hy het* **bly** *praat.*
he have:PRES stay:INF talk:INF
'He kept on talking.'

(c) *Hy het* **gebly** *praat.*
he have:PRES stay:PP talk:INF
'He kept on talking.'

While Dutch and German grammars mention general types of verbs (e.g. modal verbs) for which IPP is either required or optional, none of our Afrikaans sources do. Nevertheless, Ponelis (1979), Zwart (2007), and De Vos (2001) report that the IPP effect appears optionally in Afrikaans. This contrasts with Dutch and German, as in those languages the IPP phenomenon is obligatory for certain verbs, see amongst others Haeseryn et al. (1997), and Dudenredaktion (2006). Donaldson (1993) mentions however that IPP is triggered in most cases, such as in example (1b). Constructions with a past participle such as (1c) do occur, but Donaldson considers them non-standard Afrikaans. A similar construction as (1c) in Dutch is not possible, as the cognate verb *blijven* (stay) always triggers IPP.

De Vos (2001) also reports that some of the IPP triggers, esp. *laat* (let), tend to passivize fairly productively (2). This phenomenon is ungrammatical in Dutch and German.

(2) *Hierdie huis is deur my oom (ge)laat bou.*
This house be:PRES by my uncle let:PRES/PP build:PRES
'My uncle had this house built.'

3.2 IPP in progressive constructions

Apart from *double infinitive* constructions, there is a second construction in which IPP can be triggered. Afrikaans has a serialization pattern using the conjunction *en* (and) in order to express the continuous or progressive aspect of the verb, as in example (3a). Such constructions also exist in English (e.g. *He sits and reads*), but not in Dutch nor German. In the perfect of this construction, the first main verb has optional *ge*-marking, so it optionally triggers IPP, while the second main verb always occurs in the infinitive, as shown for the verb *staan* (stand) in examples (3b-c). Both forms are considered standard Afrikaans by Ponelis (1979), Zwart (2007), Donaldson (1993), and Verdoolaege and Van Keymeulen (2010).

[5] Some verbs have no *ge*-prefix though, so the past participle might actually be the same as the infinitive, e.g. *bestuur* (drive), *begin* (start, begin).

[6] Note that both examples (1b) and (1c) are grammatical in Afrikaans.

(3) (a) *Ons staan stil en luister.*
we stand:PRES still and listen:PRES
'We are standing and listening.'

(b) *Ons het stil* **staan** *en luister.*
we have:PRES still stand:INF and listen:INF
'We were standing and listening.'

(c) *Ons het stil* **gestaan** *en luister.*
he have:PRES still stand:PP and listen:INF
'We were standing and listening.'

De Vos (2001) reports that, although speaker judgments might vary, it is generally difficult to passivize indirect linking verbs (4), while Breed (2012) considers them grammatical.

(4) *Die appel word deur hom gesit en eet.*
The apple become:PRES by him sit:PP and eat:PRES
'The apple was being eaten by him.'

This construction is also impossible in both Dutch and German.

4 Hypothesis, data, and methodology

Based on the literature, the hypothesis is that, in contrast to Dutch and German, IPP occurs optionally in Afrikaans. We will test the hypothesis through a corpus-based study, using a PoS-tagged version of the *Taalkommissie* corpus.[7] The corpus, which is compiled by the Afrikaans language committee of the South African Academy for Science and Arts, contains about 58 million words of formal, written Afrikaans. It comprises many different text types, including newspaper articles, magazines, Bible texts, scientific articles, and study guides.

In order to query the corpus, we have created a corpus search tool (cf. section 2.4), which enables us to look for IPP constructions and their counterexamples with a past participle. We aim to find out whether IPP is actually optional or required in both double infinitive constructions and progressive constructions. Furthermore, we will investigate which verbs occur as IPP triggers in Afrikaans. The results of the corpus study are presented in section 5.

5 Results and discussion

5.1 IPP in double infinitive constructions

In order to retrieve IPP in double infinitive constructions and counterexamples with past participles in the *Taalkommissie* corpus, we extracted all combinations in which the verb form *het* (have) was followed or preceded by two verbs.[8] In addition, we also look at the sequence where there is one other word between *het* and the two other verbs. Although it is possible that more than one word occurs between *het* and the verbal group, we limited our research to constructions with zero or one word between *het* and the two verb forms.[9] This

[7] Taalkommissie van die Suid-Afrikaanse Akademie vir Wetenskap en Kuns (2011).

[8] We used the query `het[VUOT] [VT*] [VT*]` to retrieve double infinitive constructions. Discontinuous constructions as well as counterexamples were found using variations of this query.

[9] Since we only have a 'flat' corpus, it is hard to retrieve discontinuous structures. Using a treebank should solve this problem.

results in 9,880 hits, which were manually checked and categorized. We threw out the false positives due to wrong tagging and cases that did not involve main verbs that are triggering an infinitive. We also ignored the modal verbs *kan* (can), *mag* (could) and *moet* (must), as in those cases it is often hard to distinguish the matrix verb from the embedded verb.

We retained 5,679 matches for the infinitive selecting verbs, of which 5,616 occur as IPP triggers (98.89% of the constructions under consideration). The results are shown in Table 2.

Verb	IPP	No IPP	Two PPs	Total	% IPP	Translation
aanhou	45	6	0	51	88.24	keep on
begin	1,454	1	0	1,455	99.93	begin
bly	270	0	1	271	99.63	stay
doen	1	0	0	1	100.00	do, make
durf	35	1	0	36	97.22	dare
gaan	853	0	0	853	100.00	go
help	110	8	0	118	93.22	help
hoor	4	0	0	4	100.00	hear
kom	645	5	12	662	97.43	come
laat	1,458	2	0	1,460	99.86	let
leer	26	7	0	33	78.79	learn/teach
loop	0	1	0	1	0.00	walk, run
maak	1	5	0	6	16.67	make, do
ophou	16	6	0	22	72.73	stop, end
probeer	564	1	0	565	99.82	try
sien	130	5	2	137	94.89	see
wil	4	0	0	4	100.00	want
TOTAL	**5,616**	**48**	**15**	**5,679**	**98.89**	

Table 2: IPP in double infinitive constructions.

Although some verbs are used rather infrequently in this construction, it is clear that in most of the cases, IPP is actually applied. Only for *maak* and the separable verbs *aanhou* and *ophou*, we see a slightly higher percentage of cases that do not have IPP. Verbs like *begin*, *bly*, *durf*, *gaan*, *help*, *hoor*, *kom*, *laat*, *probeer*, and *sien* seem to require IPP, cf. examples (5) and (7a), while we could consider it optional at least for *leer*, cf. example (6). We also see a few cases, esp. for *kom*, in which both the main verb and the verb triggered by it appear as past participles, cf. example (7b). This is not allowed by any of the Afrikaans grammars we consulted (cf. section 3.1). In general, we can conclude that there is a clear tendency for infinitive-selecting verbs to trigger IPP. We have only found 63 sentences in which the selecting verb receives *ge*-marking, which might explain why Donaldson considers such constructions substandard. De Vos (2001) links the optionality to the level of formality.

(5) *My maag het* **begin** *draai.*
my stomach have:PRES begin:INF turn:INF
'My stomach has started to turn. [TKK, a00-2487]'

(6) (a) *Hoe ek* **leer** *lees het, weet ek nie.*
how I learn:INF read:INF have:PRES know I not
'I do not know how I have learned to read.' [TKK, a21-26482]

(b) *Dink terug hoe jy* **geleer** *bestuur het (...)*
think back how you learned:PP drive:INF have:PRES
'Think about the time you learned to drive (...)' [TKK, a16-20128]

(7) (a) *(...) Ons het* **kom** *kuier.*
we have:PRES come:INF visit:INF
'(...) We came to visit.' [TKK, a26-9964]

(b) *'n Vragmotor wat in die teenoorgestelde rigting aangery*
a lorry which in the opposite direction drive-towards:PP
gekom *het (...)*
come:PP have:PRES
'A lorry which came from the opposite direction (...)' [TKK, a44-12672]

As De Vos (2001) claimed, we found some passivized constructions with these selecting verbs (see Table 3), but they are far less frequent than the active variant. We investigated both the present form with *word* and the perfect form with *is*. Of the selecting verbs used in the passive, *laat* is by far the most frequent. There is only one counterexample using a past participle instead of the IPP construction.

Verb	IPP present	No IPP present	IPP perfect	No IPP perfect	Total	% IPP	Translation
begin	2	0	5	0	7	100.00	begin
help	0	0	1	0	1	100.00	help
laat	11	1	43	0	55	98.18	let
probeer	8	0	3	0	11	100.00	try
TOTAL	**21**	**1**	**52**	**0**	**74**	**98.65**	

Table 3: IPP in passive double infinitive constructions.

5.2 IPP in progressive constructions

In a second test, we looked at IPP triggers in the serialized form of progressive constructions.[10] We again selected cases with *het*, but now with the conjunction *en* (and) between the two content verbs. This resulted in 1,743 hits, which were again categorized manually. We only retained 244 positive examples, of which 50.82% appeared as IPP triggers. The results are shown in Table 4.

It is clear that the construction as such is only frequent using *lê*, *sit* and *staan* as IPP triggers. IPP occurs in slightly less than half of the cases for *sit* and *staan*, so we can agree with the grammars that IPP is optionally triggered in progressive constructions, cf. example (8). For *lê* however, there seems to be a clear preference for the IPP construction. The progressive also occurs a few times with *loop*, but in that case the past participle seems to be preferred. We encounter again a few cases of two past participles, cf. example (9b). Similar to the constructions with double participles in section 5.1, such constructions seem less preferred. If we compare the results with the frequencies of a verb being the trigger for the progressive construction in this corpus (Breed, 2012), we see that verbs using the progressive frequently

[10] We used the query `het[VUOT] [VT*] en[KN] [VT*]` to retrieve double infinitive constructions. Discontinuous constructions as well as counterexamples were found using variations of this query.

(*sit*, *staan*, and *lê*) are more likely to apply IPP then *loop*, which is less likely to occur in this construction.

Verb	IPP	No IPP	Two PPs	Total	% IPP	Translation
bly	0	0	1	1	0.00	stay, remain
bystaan	0	1	0	1	0.00	stand near
kom	0	0	1	1	0.00	come
lê	30	5	0	35	85.71	lie
loop	1	4	1	6	16.67	walk, run
rondstaan	0	1	0	1	0.00	stand around
sit	48	58	1	107	44.86	sit
staan	45	47	0	92	48.91	stand
TOTAL	**124**	**116**	**4**	**244**	**50.82**	

Table 4: IPP in progressive/continuous constructions.

Note that most of the verbs that use this construction do not occur in the double infinitive construction (cf. Table 2), while their Dutch cognates do (e.g. Afrikaans *lê* vs. Dutch *liggen* (to lie)). We can conclude that both constructions are in general mutually exclusive.

(8) (a) *(...) waar hy die spul onder 'n soetdoring* **sit** *en dophou*
where he the stuf under a sweet thorn tree sit:INF and watch:INF
het (...)
have:PRES
' (...) where he was watching the stuff under a sweet thorn tree (...)' [TKK, a25-14908]

(b) *Ek het daar* **gesit** *en wag op Brett (...)*
I have:PRES there sit:PP and wait:INF for Brett
'I was waiting there for Brett (...)' [TKK, a34-8014]

(9) (a) *Hy vertel hoe hy (...) vir die hysbak* **staan** *en wag*
he tell:PRES how he for the lift stand:INF and wait:INF
het (...)
have:PRES
'He tells how he (...) waited in front of the lift (...)' [TKK, a34-1063]

(b) *(...) 'n paar meter van waar ek nog so rustig* **gestaan** *en gesels*
a couple metre from where I still so quiet stand:PP and chat:INF
het.
have:PRES
'(...) a couple of metres from where I was chatting (...)' [TKK, a34-1086]

According to Breed (2012) passive constructions with indirect linking verbs are possible, but she was, like us, not able to find any examples in the *Taalkommissie* corpus.

6 Conclusions and future work

The case study on IPP triggers in Afrikaans shows that a corpus-based study can shed a new light on the descriptive research of a linguistic phenomenon. Based on the literature, we assumed that IPP is optionally triggered in Afrikaans (both in double infinitive constructions

and in progressive constructions). The corpus results, however, reveal that infinitive-selecting verbs in double infinitive constructions trigger IPP in almost 99% of the constructions under investigation. The results of the progressive constructions are more consistent with the current literature, since the IPP phenomenon optionally occurs in such constructions (i.e. in ca. 50% of the cases). Moreover, we can conclude that verbs that occur as IPP triggers in the double infinitive construction, do not occur as IPP triggers in the progressive construction and vice versa.

Although we obtained some nice results from the present study, we had to do a lot of (semi-)manual filtering of the results. In order to reduce such tasks, as well as to improve the quality of the annotated data, we will improve the output of the annotation tools in future research. As the CTexT tagger still contains a lot of errors which could be corrected by a simple rule-based extension, we will create a rule-based tag corrector based on the Brill tagger (Brill, 1992).

We also want to extend the parser in order to have different options from the current shallow parsing (including updating and improving the current grammars) to a full parse tree. The parser will then be used to find constructions with more tokens intervening between the relevant items (i.e. between the auxiliary *het* and the infinitives and past participles in our case study). We will need to adapt the search tool to be able to search for chunks as well. Of course, this includes dealing with issues like efficient querying, indexing, and the representation of the trees in the tool. Besides improving existing tools, we will run a lemmatizer on the data, in order to include lemmas in the search tool as well.

Finally, all of this will be integrated in an Afrikaans equivalent of GrETEL (Augustinus et al., 2012), a query engine in which linguists can use a natural language example as a starting point for searching a treebank with limited knowledge about tree representations and formal query languages.

Using all these tools, we want to further investigate the IPP effect in Afrikaans. For example, it would be interesting to investigate whether the number of tokens occurring between the auxiliary and the other verb(s) have an influence on the construction used. Those results can also be useful for a cross-linguistic comparison with similar work in Dutch and German.

Acknowledgments

We wish to thank the people of the Taalkommissie and CTexT for providing us with the *Taalkommissie* corpus and the PoS tagger.

References

Augustinus, L., Vandeghinste, V., and Van Eynde, F. (2012). Example-Based Treebank Querying. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul.

Brants, T. (2000). TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP-2000)*, pages 224–231, Seattle.

Breed, A. (2012). *Die grammatikalisering van aspek in Afrikaans: semantiese studie van perifrastiese progressiewe konstruksies*. PhD thesis, North-West University, Potchefstroom.

Brill, E. (1992). A simple rule-based part of speech tagger. In *Proceedings of the third conference on Applied natural language processing (ANLC 42)*, pages 152–155, Stroudburg, PA.

De Vos, M. (2001). Afrikaans Verb Clusters: A Functional-Head Analysis. Master's thesis, University of Tromsø, Tromsø.

Dirix, P., Vandeghinste, V., and Schuurman, I. (2005). METIS-II: Example-based machine translation using monolingual corpora – System description. In *Proceedings of MT Summit X, Workshop on Example-Based Machine Translation*, pages 43–50, Phuket.

Donaldson, B. C. (1993). *A Grammar of Afrikaans*. Mouton de Gruyter, Berlin/New York.

Dudenredaktion (2006). *DUDEN. Die Grammatik. Unentbehrlich für richtiges Deutsch*. Dudenverlag, Mannheim/Leipzig/Vienna/Zürich.

Grover, A. S., van Huyssteen, G. B., and Pretorius, M. W. (2011). A Technology Audit: The State of Human Language Technologies (HLT) R&D in South Africa. In *Proceedings of PICMET'11: Technology Management In The Energy-Smart World (PICMET)*, pages 1693–1706.

Haeseryn, W., , Romijn, K., Geerts, G., de Rooij, J., and van den Toorn, M. (1997). *Algemene Nederlandse Spraakkunst*. Martinus Nijhoff/Wolters Plantyn, Groningen/Deurne, second edition.

Pilon, S. (2005). Outomatiese Afrikaanse woordsoortetikettering. Master's thesis, North-West University, Potchefstroom.

Ponelis, F. A. (1979). *Afrikaanse Sintaksis*. J.L. van Schaik, Pretoria.

Puttkammer, M. J. (2006). Outomatiese Afrikaanse tekseenheididentifisering. Master's thesis, North-West University, Potchefstroom.

Schlünz, G. I. (2010). The effects of part-of-speech tagging on text-to-speech synthesis for resource-scarce languages. Master's thesis, North-West University, Potchefstroom.

Taalkommissie van die Suid-Afrikaanse Akademie vir Wetenskap en Kuns (2011). Taalkommissiekorpus 1.1., CTexT, North West University, Potchefstroom.

Vandeghinste, V. (2008). *A Hybrid Modular Machine Translation System*. PhD thesis, University of Leuven.

Verdoolaege, A. and Van Keymeulen, J. (2010). *Grammatica van het Afrikaans*. Academia Press, Ghent.

Zwart, J.-W. (2007). Some notes on the origin and distribution of the IPP-effect. *Groninger Arbeiten zur Germanistischen Linguistik*, 45:77–99.