Originally published in: Southern African Linguistics and Applied Language Studies vol. 26 (2008) no. 2, pp. 185-196.
DOI: *https://doi.org/10.2989/SALALS.2008.26.2.1.565*

POSTPRINT

Danie J Prinsloo*
Gertrud Faaß
Elsabé Taljard
Ulrich Heid[1]

Department of African Languages, University of Pretoria, Pretoria 0002, South Africa
[1]*also Institut für maschinelle Sprachverarbeitung – Computerlinguistik, Universität Stuttgart, Azenbergstrasse 12, D–70174 Stuttgart, Germany*
**corresponding author, e-mail: danie.prinsloo@up.ac.za*

Designing a verb guesser for part of speech tagging in Northern Sotho

Abstract: The aim of this article is to describe the design and implementation of a verb guesser that will enhance the results of statistical part of speech (POS) tagging of verbs in Northern Sotho. It will be illustrated that verb stems in Northern Sotho can successfully be recognised by examining their suffixes and combinations of suffixes. Two approaches to verbal derivation analysis will be utilised, namely morphological analysis and corpus querying of suffixes and combinations of suffixes.

Introduction

Future technological developments such as spelling checkers, grammar checkers and machine translation systems for African languages in general and Northern Sotho in particular, are heavily dependent on corpora that are annotated with part of speech (POS) tags. The main challenge to POS tagging for African languages lies in automatic tagging of the so-called open classes of which the main ones are verb stems and nouns. Formulated in simple terms, the main issue regarding verb and noun guessing is to devise a methodology that can maximally utilise the morphological and syntactic features of verb stems and nouns in order to automatically identify and tag them in any given text. Our approach for verb guessing in Northern Sotho will be the identification and description of the typical morphological and syntactic features of verbs, and the utilisation of such features as a tool to identify verb stems. The discussion will be concluded by a quantitative evaluation of the automatic guessing of verb stems. At this stage the degree of specificity of the tagger for verbs is limited to the identification of verb stems only and to tag them as V; no finer-grained tagset is implemented.

The correct word class of a token can be determined by either statistical or rule-based methods, with a hybrid approach being a third possibility. Statistical methods, as described in De Schryver and De Pauw (2007) use training data to determine the correct annotation of a certain token in a certain environment. Provided that the tagset is not too extensive, and the text type similar to that of the training data, this approach delivers good results (overall more than 90% accuracy). These results cause rule-based and even hybrid approaches to be regarded as rather outdated. In case of an extended tagset and/or lack of sufficient training data, hybrid approaches do however promise good results (see Prinsloo & Heid, 2006). However, it needs to be noted that the accuracy with which such methods determine the correct class of unknown words, that is, words belonging to productive classes that do not occur in the training data at all, is substantially reduced: according to De Schryver and De Pauw (2007: 240) the statistical approach reaches an accuracy of 78.9% for unknown words. The verb guesser described in this article aims at specifically identifying such words. As it is part of a guessing process that also identifies nouns (and, possibly at a later stage, adverbs), this tool helps to identify all possible annotation(s) of yet unknown tokens, and thus can act as a preprocessing step to a statistical POS-tagging system enhancing its results.

This tool could also provide data for enhancing the lexical database of, for example, morphological analysers (see Kotzé & Anderson, 2006) in delivering fully inflected verb stems used in the language.

The basic methodology for verb guessing is firstly based on the detection of typical verbal suffixes and combinations of such suffixes. Secondly, as verbal affixation often involves morphophonological changes, in some instances rules reversing these sound changes will have to be applied before the suffixes are identified. Thirdly, an analysis-by-generation approach will be used: frequently occurring

verbal affixes will be attached to the verb stem candidates. These generated forms will then be checked against an existing word form lexicon annotated with POS and those not found in this lexicon will be searched for in the seven-and-a-half million word University of Pretoria Sepedi Corpus (PSC) (see De Schryver & Prinsloo, 2000). The system lexicon contains verbs in about 3 700 forms, nouns, adverbs, concords, pronouns, adjectives, morphemes, particles, question words and others.

The discussion on verb guessing is preceded by an analysis of verbal suffixes as described in standard Northern Sotho grammars. The analysis of possible series of combinations of verbal suffixes will be based on derivations listed for 300 randomly selected verbs from Ziervogel and Mokgokong's *Groot Noord-Sotho woordeboek* (1975), henceforth *GNSW*.

Verbal derivations in Northern Sotho

A core issue in the design of the verb guesser is to determine which suffixes and which combinations of suffixes – termed suffix clusters – occur in Northern Sotho. As a starting point, grammatical descriptions of verb stems and verbal suffixes in sources such as Lombard (1985), Louwrens (1991), Poulos and Louwrens (1994), and Van Wyk *et al*. (1992), were consulted. Information thus gleaned was supplemented by corpus data, as well as the comprehensive verbal derivation paradigms given in *GNSW*. Current views on verbal suffixes as found in the afore-mentioned sources are summarised in Table 1.

The discussions in grammar books mentioned above focus mainly on listing and illustrating the meaning and use of the different suffixes and do not go into any detail regarding the combinations

Table 1: Verbal suffixes in Northern Sotho

Name	Abbr	Form	Example	Translation
Neuter	Neut	*-agal-*	*bona*	see
			bonagala	be visible
Associative	Ass	*-agan-/*	*aba*	divide, distribute
		-akan-	*abagana*	divide amongst one another
Iterative	Ite	*-ak-*	*bofa*	tie
			bofaka	tie over and over again
Neutro-active	NAct	*-al-*	*bona*	see
			bonala	be visible
Dispersive	Dis	*-alal-*	*falala*	overflow, become scattered
			falalala	disperse
Positional	Pos	*-am-*	*uta*	hide (intransitive/transitive)
			utama	lie flat
Reciprocal	Rec	*-an-*	*rata*	love
			ratana	love each other
Alternative causative	ACau	*-ny-*	*fapana*	differ
			fapanya	distinguish
Contactive	Con	*-ar-*	*apara*	clothe
Neutro-passive	NPas	*-eg-*	*rata*	love
			ratega	be lovable
Applicative	App	*-el-*	*reka*	buy
			rekela	buy for
Perfect	Per	*-il-*	*dira*	do, make
			dirile	did, made
Causative	Cau	*-iš-*	*ruta*	learn
			rutiša	teach
Passive	Pas	*-iw-/*	*thuša*	help
		-w-	*thušwa*	be helped
Reversive intransitive	RInt	*-og-/ -olog-*	*tla*	come
			tloga	go away
Reversive transitive	RTra	*-ol-/-oll- /-olol-*	*bofa*	tie
			bofolla	untie, loosen

of suffixes. *GNSW*, in terms of its lemmatisation strategy, attempts to enter all possible derivations of each individual verb and therefore supplies valuable information on:

- verbal derivations, not only for the target user of the dictionary, but also for the computational linguistic studies described in this article; and
- patterns for the implementation of a morphological analyser or generator, that is, a morphology which will generate all possible surface forms – with such a morphology module at hand, a full form lexicon will become dispensable (on the basis of a lexicon of verbal roots combined with rules describing the formation of verbs).

Compare the analysis of the verb *gadika* 'roast, thrash' as found in *GNSW*, tabulated in Table 2 as a typical example.

The schematic presentation in Table 2 is subdivided into 7 groups, each consisting of the verbal root *-gadik-* with either a suffix or a suffix cluster. Apart from the verb stem *gadika* listed first, the 25 derivations distinguished in *GNSW* consist of (clusters of) 6 suffixes: perfect, passive, reciprocal, applicative, causative and neutro-passive. For the verb stem *bona* 'see', *GNSW* distinguishes altogether 24 'modules' with 86 derivations. The term module refers to the standard modifications described in Table 2.

In terms of developing a morphological analyser, what first needs to be determined is the full paradigm of all possible combinations of suffixes. If such a paradigm could be compiled, it could form the core element of the module for verb guessing, because a possible verbal status of any given token could then be determined with a high probability. Such a module could also be used as part of a lemmatiser, consequently not only identifying a verb stem as such, but also the derivational suffixes used, thus providing necessary information for further processing steps, for example, a semantic analysis as part of a machine translation system.

In theory, it is assumed that the suffixes that are shown in Table 1 can combine in many ways, forming clusters. In order to create an inventory of such clusters to be used as input data for the verb guesser, a random selection of 300 verb stems was made and their verbal derivations were extracted from *GNSW*, for example, *gadika* in Table 2. The verbs selected are given in Appendix 1. The list of suffixes and combinations of suffixes (suffix clusters) that *GNSW* distinguishes for these verbs is fully described in Appendix 2. At the current state, 350 suffixes and suffix clusters have been identified. However, as this is work in progress, the number is expected to grow.

Another feature of all verbal suffixes and suffix clusters that is relevant to the guessing procedure is their ability to combine with the relative suffix *-go*. The affixation of this suffix to those listed in Appendix 2 would result in forms such as *-ago*, *-aditšego*, *-aditšwego*, *-afaditšego*, *-afaditšwego*, *-afalago*, etc. For these experiments the verb guesser was instructed to delete the suffix *-go* before any candidate's suffix is compared with the list of 350 suffixes. If the remaining suffix is found in the list, the candidate qualifies for further testing. In future, the presence of *-go* as a part of the suffixal string could be considered in verbal detection because it could serve as additional criteria for verb guessing. However, this testing will not in all cases be necessary, as the respective character strings are only found as verbal suffix clusters (see Appendix 2). Therefore candidates containing these strings can immediately be identified as verbs.

Strategies for verb guessing

The longest match principle

The suffixes listed in Appendix 2 are utilised in two ways in the process of guessing. Firstly, words are tested against these suffixes on a longest-match basis. The underlying assumption is that the longer the matching suffixal substring, the higher the probability of successful verb guessing would be. Consider the following results of a corpus query on the PSC for words ending in *-išeditšwe* and *-ela*. In the case of *-išeditšwe,* all are indeed verbs. See Table 3 in this regard. These results indicate that the probability of a non-verb displaying the exact formation of the nine-letter affixal cluster *-išeditšwe* is extremely low.

The short substring *-ela* would be far less efficient as sole criterion for verb guessing. Although many verbs will be guessed correctly, nouns such as *tsela/ditsela* 'road(s)', *mosela/mesela* 'tail(s)', *setlaela/ditlaela* 'dumb person(s)', *botlaela* 'stupidity', *setimela/ditimela* 'train(s)', *lešela/mašela*

Table 2: Derivations of the verb *gadika* ‘roast, thrash’ in GNSW

Suffix: -A

	root + standard modifications			
Structure	ROOTa	ROOTile	ROOTwa	ROOTilwe
Grammatical formula	VR	VRPer	VRPas	VRPerPas
Example	*Gadika*	*Gadikile*	*Gadikwa*	*Gadikilwe*
Translation	roast/thrash	roasted/thrashed	be roasted/thrashed	was/were roasted/ thrashed

Suffix: -ANA

	root + reciprocal + standard modifications			
Structure	ROOTana	ROOTane	ROOTanwa	ROOTanwe
Grammatical formula	VRRec	VRRecPer	VRRecPas	VRRecPerPas
Example	*Gadikana*	*Gadikane*	*Gadikanwa*	*Gadikanwe*
Translation	roast/thrash each other	roasted/thrashed each other	(theoretical form)	(theoretical form)

Suffix: -EGA

	root + neutro-passive + standard modifications			
Structure	ROOTega	ROOTegile		
Grammatical formula	VRNPas	VRNPasPer	(VRNPasPas)	(VRNPasPerPas)
Example	*Gadikega*	*Gadikegile*		
Translation	be roasted/ thrashed	was/were roasted/ thrashed		

Suffix: -ELA

	root + applicative + standard modifications			
Structure	ROOTela	ROOTetše	ROOTelwa	ROOTetšwe
Grammatical formula	VRApp	VRAppPer	VRAppPas	VRAppPerPas
Example	*Gadikela*	*Gadiketše*	*Gadikelwa*	*Gaditketšwe*
Translation	roast for	roasted for	be roasted for	was/were roasted for

Suffix: -ELANA

	root + applicative + reciprocal + standard modifications			
Structure	ROOTelana	ROOTelane	ROOTelanwa	ROOTelanwe
Grammatical formula	VRAppRec	VRAppRecPer	VRAppRecPas	VRAppRecPerPas
Example	*Gadikelana*	*Gadikelane*	*Gadikelanwa*	*Gadikelanwe*
Translation	roast for each other	roasted for each other	be roasted for each other	was/were roasted for each other

Suffix:-IŠA

	root + causative + standard modifications			
Structure	ROOTiša	ROOTišitše	ROOTišwa	ROOTišitšwe
Grammatical formula	VRCau	VRCauPer	VRCauPas	VRCauPerPas
Example	*Gadikiša*	*Gadikišitše*	*Gadikišwa*	*Gadikišitšwe*
Translation	cause to thrash	caused to thrash	cause to be thrashed	caused to be thrashed

Suffix:-IŠANA

	root + causative + reciprocal + standard modifications			
Structure	ROOTišana	ROOTišane	ROOTišanwa	ROOTišanwe
Grammatical formula	VRCauRec	VRCauRecPer	VRCauRecPas	VRCauRecPerPas
Example	*Gadikišana*	*Gadikišane*	*Gadikišanwa*	*Gadikišanwe*
Translation	cause to thrash each other	caused to thrash each other	cause each other to be thrashed	caused each other to be thrashed

‘cloth(s)’ end in a string which is homographous with this suffix, and (like many other nouns) would therefore be incorrectly guessed as verbs should the suffix *-ela* be the sole criterion. Moreover, the string *-ela* could even be part of a verb stem, as it is in *hlasela* ‘attack’. Therefore, it is clear that string matching, although very effective, should be supplemented by additional strategies to enhance guessing procedures for verbs.

Table 3: Types ending in *-išeditšwe* in the PSC

Example	Translation was/were …	Example	Translation was/were …
dirišeditšwe	used for	*kgonthišeditšwe*	justified for
emišeditšwe	made to stand up for	*lokišeditšwe*	repaired for
fetišeditšwe	prepared for	*phadimišeditšwe*	made to shine for, was/were polished for
fihlišeditšwe	delivered to / be hidden from	*šutišeditšwe*	made to be moved up for
fišeditšwe	burnt for	*tiišeditšwe*	strengthened for
gatišeditšwe	printed for	*tlišeditšwe*	brought for
hlagišeditšwe	created for	*tsebišeditšwe*	informed for/of
ikemišeditšwe	prepared for	*tšhabišeditšwe*	made to flee from

The reversal of sound changes

Though Northern Sotho is considered to be a disjunctively written language, there are instances where morphemes (other than the verbal suffixes) are orthographically conjoined to the verbal root. In some cases, such affixation of prefixal morphemes to verbal roots causes phonological changes, such as plosivation, elision and assimilation. This implies that the initial letter(s) of the root – as orthographical representations of speech sounds – might change when a prefixal morpheme is added to the root. Of particular relevance for this discussion are the phonological changes caused by prefixing of the:

(a) nasal object concord, first person (*N-*);
(b) object concord class 1 *mo-;* and
(c) reflexive morpheme *i-*.

See Table 4 for examples of these changes.

Generally, one could expect a candidate beginning with *i-* to be unambiguously verbal, because there is no nominal prefix in Northern Sotho beginning with this letter. All other words with initial *i-* belong to closed classes, that is, they are listed in the system's lexicon. On the other hand, as this language is developing, a number of loan words starting with *i-* have been introduced, for example, *intasteri* 'industry' and *inšorentshe* 'insurance' which are indeed nouns. More of such loan words are expected to become a regular part of future corpora. It was hence decided to also subject such candidates to further testing.

In the case of reflexive formation, the general rule is that the reflexive prefix *i-* is merely prefixed to the verb stem. However, for most verb stems, plosivation will cause the initial consonant to change. Therefore, cutting away the prefix in cases such as in Table 4(c) will not result in a positive guess, because this procedure stops short of determining the correct stem. For this particular example, the verb guesser would erroneously consider *tlhompha* and *thata* as possible verb candidates, instead of the correct forms, that is, *hlompha* and *rata* respectively.

The phonological changes given in Table 5 therefore have to be reversed in order to find the correct verbal stems. This would obviously also apply to all other forms containing the reflexive

Table 4: Examples of phonological changes

Nature of prefix	Verb stem	Resultant form	Phonological change
a) Object concord – 1st person singular			
n-	*Šomela*	*ntšhomela* 'work for me'	plosivation
m-	*Fa*	*mpha* 'give me'	assimilation & plosivation
b) Object concord – class 1			
mo-	*Bitša*	*mmitša* 'him/her call'	elision & assimilation
c) Reflexive morpheme			
i-	*Hlompha*	*itlhompha* 'respect oneself'	plosivation
i-	*Rata*	*ithata* 'love oneself'	plosivation

prefix and the object concord of the first person singular (*N-*). Consider examples of sound changes for the object concord of the first person singular and the reflexive in Tables 5 and 6 respectively. In the case of the object concord of class 1, only a single rule applies, that is, *mo-* + *b-* > *mm-*, thus *mo+bitša* 'him/her call' > *mmitša*.

Testing roots using frequent endings

For word forms beginning with *i-* and forms containing object concords, considering only their basic form, plus the applicative and/or the causative form could be the decisive factor in successful verb guessing. Consider the example below where (a) shows the verb stem plus an object concord (first person singular) prefixed to the stem, and (b) shows the basic stem form with the applicative and causative suffixes:

(a) ntsoma (8), ntsomela (0), ntsomiša (0); and
(b) tsoma (685), tsomela (12), tsomiša (6).

The numbers in parenthesis show the frequency in the PSC.

Checking only for the form *ntsoma* would not be sufficient, as any candidate should occur in at least one other verbal derivative form to prove its verbal status. The verbal status of *ntsoma* cannot be confirmed, because neither the applicative *ntsomela* nor the causative *ntsomiša* occurs in the PSC. Only after the sound change has been reversed, and the applicative and causative suffixes added, can the corpus query render hits.

The same procedure would be followed for cases where the object concord of class 1, *mo-* is prefixed to verbs with initial *b-*, causing the sound change as explained above.

Table 5: Examples of sound changes caused by the object concord of the first person singular

Object conc. (1sg) + Vstem	Translation	Result of sound changes
n + *a*raba	'me + answer'	> n*k*araba… (a > ka)
n + *b*ona	'me + see'	> *mp*ona... (b > p and ...nb > mp)
n + *d*irela	'me + work for'	> n*t*irela... (d > t)
n + *e*tela	'me + visit'	> n*k*etela... (e > ke)
n + *f*ora	'me + deceive'	> *mph*ora... (f > ph and ...nf > mph)
n + *g*opotša	'me + remind'	> n*k*gopotša… (g > kg)
n + *h*umiša	'me + enrich'	> n*k*humiša… (h > kh)
n + *hl*ompha	'me + respect'	> n*tlh*ompha…(hl > tlh)
n + *i*ša	'me + take'	> n*k*iša... (i > ki)
n + *j*eša	'me + feed'	> n*tš*eša… (j > tš)
n + *k*gopela	'me + ask'	> nkgopela... (no change)
n + *l*oma	'me + bite'	> n*t*oma... (l > t)

Table 6: Examples of sound changes caused by the reflexive morpheme

Reflexive + Vstem	Translation	Result of sound changes
i + *a*raba	'self + answer'	> i*k*araba... (a > ka)
i + *b*ona	'self + see'	> i*p*ona... (b > p)
i + *d*irela	'self + work for'	> i*t*irela... (d > t)
i + *f*ora	'self + deceive'	> i*ph*ora... (f > ph)
i + *g*opotša	'self + remind'	> i*k*gopotša… (g > kg)
i + *h*umiša	'self + enrich'	> i*k*humiša… (h > kh)
i + *hl*ompha	'self + respect'	> i*tlh*ompha…(hl > tlh)
i + *i*ša	'self + take'	> i*k*iša... (i > ki)
i + *j*eša	'self + feed'	> i*tš*eša… (j > tš)
i + *k*gopela	'self + ask'	> ikgopela... (no change)
i + *l*oma	'self + bite'	> i*t*oma... (l > t)

Implementation of the verb guesser

The verb guesser is implemented as a staged system which tries to identify roots for each input item – firstly (A) and (B), and then proceeds from simple to more complex cases in four stages from (C) to (F). See Figure 1 for the sequence of modules in implementing a verb guessing process.

(*A) Getting input candidates*

A list of the 350 verbal suffix clusters described in Appendix 2 is used as linguistic knowledge. The relative suffix *-go* can be added to all of them. When a tokenised file, that is, a file in a one-token-per-line format, is offered to the guessing module, the guessing procedure will firstly check for tokens ending in one of these suffixes, as shown in Figure 1, module (A). The processing only continues if this is the case.

(*B) Root identification*

As some suffixes are included in suffixal strings (for example, *-ela* in *-ollelana*) there may be several possibilities when trying to identify the correct root for some candidates. For each of these, a list is created with all possible roots and their suffixes. This list is then sorted from the longest to the shortest suffix in order to try the longest match first. For example, the verbal suffix *-agalela* comprises *-ela* and also *-a*. As the verb guesser is to follow the longest match strategy, looping through all possible suffixes, it would examine (see module (B) in Figure 1) candidates like *diragalela* 'befall, occur to, happen to, experience', three times. First, it would try the longest match *-agalela*, resulting in the root *dir-*. As a second step it would do the tests described in the next paragraph with the second longest suffix, *-ela* (root *diragal-)*. Finally, the verbal ending *-a* would be taken as last possibility, assuming the root of the candidate is *diragalel-*. Module (B) also reverses sound changes, where applicable.

(C) Identifying purely verbal candidates

As described above, most of the suffix cluster strings do only occur as verbal suffixes. Further tests on verb candidates containing such suffixes, as described in (D) and (E) below, would be done unnecessarily in most cases – a candidate like *diragalela* can be identified with a high probability as a verb, using only its suffixal string as criterion. Therefore, clusters like *-agalela* are marked in the list of verbal suffixes as being 'safe' verb form indicators (compare items in roman print in Appendix 2). If a candidate shows such a suffix, module (C) immediately stops the guessing process and suggests it as a verb. However, as already pointed out, there are indeed verbal roots that include patterns identical to suffixes, for example, *hlasela* 'attack' where *-el-* is not an applicative suffix, but part of the root.

For substrings like *-ela*, the guesser thus must loop through all possible root-suffix combinations; otherwise the correct root *hlasel-* of the verb *hlasela* would not be identified. Additionally, as described above, there are a number of nouns which end in strings coincidentally homographous with verb suffixes; therefore these affixes are marked in the list of verbal suffixes as possibly ambiguous in terms of their part of speech (see items in bold italics in Appendix 2).

(*D) and (E) Testing the candidates*

If a candidate has been identified as potentially verbal, based on its suffixal structure, and if its suffix could be ambiguous, the verb guesser is instructed to add the frequent suffixes *-a*, *-ela* and *-iša* to the putative root. Thereafter module (D) will check if one of the forms thus created occurs in the system's lexicon. Consider the candidate *rekišitšwe* 'was sold' as an example. The longest possible suffix contained in the list of verbal endings is *-išitšwe,* hence this suffix will be cut off and the assumed root *rek-* and the assumed base forms *reka* 'buy'*, rekela* 'buy on behalf of', and *rekiša* 'sell' will be generated. If one of these forms occurs in the system's lexicon as a verb, module (D) ends the guessing process and classifies the candidate *rekišitšwe* as a verb.

Should the generated forms *reka*, *rekela* and *rekiša* not be found in the lexicon of the system, module (E) will check the PSC for occurrences of these forms. In order to reach a high precision, a threshold was implemented: at least two of the generated three forms have to be found in the corpus.

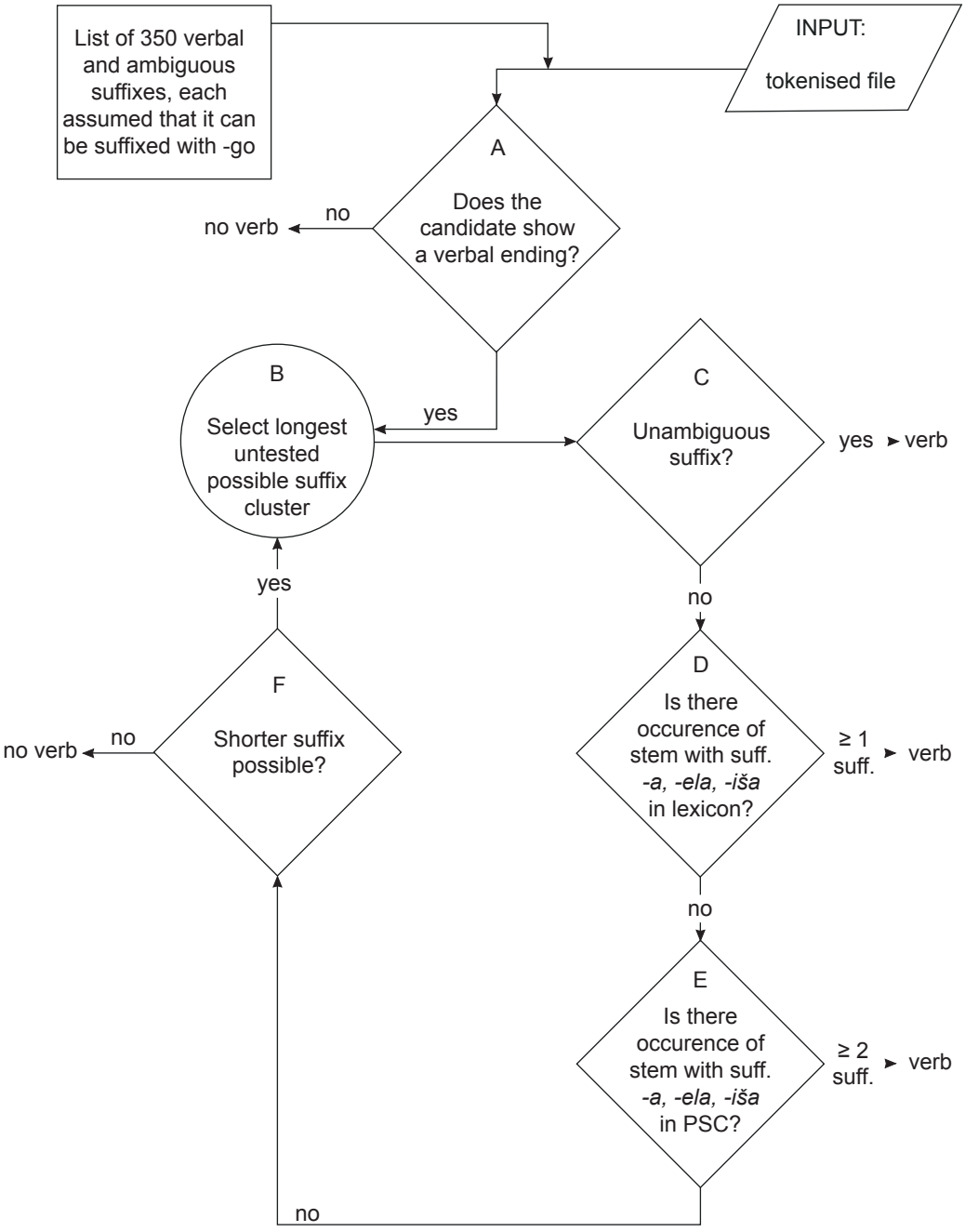

Figure 1: The modules of the verb guessing process

(F) Checking the remaining list of candidates

Module (F) only checks if there are still possible root-suffix combinations left in the list created by module (B). If there are, the next entry of the list is submitted to the loop (C)-(D)-(E), and subsequently (F) again. When no more combinations are left, the guesser exits.

Dealing with ambiguity

Candidates whose verbal suffixal strings are homographous with endings of nouns, will be tested by means of a second guessing module for nouns (see Heid *et al.*, forthcoming) irrespective of whether the verb guessing was successful or not. As the noun guesser identifies nouns rather generously, a candidate might as a result be guessed ambiguously as a verb and as a noun. Consider the verbal candidate *aga* 'build, live', which is guessed as V or N05 or N09 by the combined guessers.

On the other hand, in the case of *kwana* (V) 'to agree' and *kwana* (N09) 'lamb', for example, verb and noun guessing are both necessary, as this is a case of homonymy. Therefore, the developers opted for a recall-oriented system which doesn't exclude *a priori* any possible analysis.

More generally, this behaviour allows for ambiguity awareness, because unambiguous cases and ambiguous ones are kept separate, and the latter submitted for human judgment.

Results and evaluation

The verb guesser was evaluated with a corpus of Northern Sotho consisting of 11 287 tokens of a thesis written by Thobakgale (2005). This text is neither part of the PSC nor has it been exploited for the system dictionary so far. This corpus contains 1 712 types, of which 619 types are not contained in the system's lexicon and hence have to be guessed. Manual classification resulted in the identification of 145 verbal types.

The verb guessing procedure resulted in 155 types guessed as verb stems, of which 136 are true positives. Nineteen candidates indeed have endings similar to verbal patterns, but are actually nouns; for example, *Kanana*, a place name, or *Podile,* a person's name (false positives). The verb guesser hence delivers a precision of 0.8774 (88%). As explained above, the procedures are recall-oriented, that is false positives are accepted.

However, the remaining correctly identified 136 verb stems did not include 9 verb forms that had been manually identified (false negatives): *amologantšhwe* 'been taken away from'*, jetše* 'ate from'*, gagara* 'kill without exception'*, kgotlopa* 'walk in mud'*, nenenkela* 'walk slowly', *ntšifala* 'multiply', *phošišwe* 'be guessed', *swaswa* 'make jokes', and *tsikiditlago* 'which tickles', and hence the verb guesser reaches a recall of 0.9379 (94%).

An interesting example is *amologantšhwe*, which contains the verbal ending *-antšhwe.* The suffix cluster *-antšhwe* is rather rare (8 types with 30 tokens in the 6 million words of the PSC) and thus was not contained in the first version of the suffix clusters described in Appendix 2, so the verb guesser was unable to identify the root *amolog-* as such. As the verb guesser signals cases where no analysis can be provided, it is possible to use it to interactively update the inventory of suffix clusters, as was done in the case of *-antšhwe.*

A rule has been formulated that informs the verb guesser that any verbal root must contain at least three characters. This rule was implemented to save processing time, due to the fact that very few roots consist of only a single character. Furthermore, there is only one form that omits the root completely, that is, *ile*, the past tense form of *ya* 'go'. All known regular forms of the one-character-roots should be listed in the tagger lexicon. One of these roots is *j-* 'eat'. In the test corpus there is the infrequent form *jetše*, which is an applicative (*-ela)* of *j-,* combined with the past tense form (*-ile*). Combining these two suffixes leads to the form *jetše,* which was not included in the system's lexicon. Due to the described rule, the verb guesser was not able to identify the correct root of *jetše*.

The verb guesser correctly identified the roots of all of the other 7 verbs: *gagara*, *kgotlopa, nenenkela*, *ntšifala*, *phošišwe*, *swaswa* and *tsikiditlago.* However, as evidence could not be found (neither in the lexicon of the system nor in the corpus) for the generated forms *gagarela, gagariša*, *kgotlopela, kgotlopiša*, etc., the guesser found no proof for its correct assumption.

In other words, only 2 out of 145 verb stems could not be identified by the guesser: for one the suffix had not yet been listed in the list of verbal endings, and for the other the full form was not listed in the system's lexicon. For the 7 other forms, no evidence was found to conclusively identify them because of the frequency threshold indicated above.

At the current state of implementation, the verb guesser delivers an *F*-measure[1] of 0.907 (91%).

---

[1] F-measure is the weighted harmonic mean of precision and recall calculated as
F = 2 × (precision × recall)/(precision + recall).

Conclusions and future work

As verbs are one of the open word classes of Northern Sotho, and since Northern Sotho verbal morphology is very rich and productive, no lexicon of verb forms for the computational processing of this language could ever be complete. To process Northern Sotho texts, there is thus a need for a procedure able to identify unseen verb forms.

For the purpose of POS tagging of Northern Sotho, we developed a verb guesser which relies on the analysis of verbal derivation affixes and especially suffix clusters. Some suffix clusters identify verbs with a very high likelihood (see Appendix 2, stems in roman font); others contain strings of characters which can also appear in Northern Sotho nouns (compare the bold italicised items in Appendix 2). We designed and implemented a staged procedure (combinable modules) to classify word forms as verbal by trying to identify the longest possible suffix clusters, repeating the procedure if necessary with shorter forms. Where necessary, morphophonological changes between verb stem and derived forms are taken into account. Cases where an ambiguity remains are cross-checked by means of analysis-by-generation: the verb root candidate is used to generate putative verb forms which are searched in the lexicon of the system and in corpus data.

The verb guesser is part of a larger attempt to provide tools to robustly POS tag Northern Sotho texts with a high precision. As such, it is used, together with a guesser for noun forms, as a pre-tagging step for statistical POS tagging (see Prinsloo & Heid (2006) for the overall architecture). Homography between verbal and nominal forms can be detected by the combined use of both guessers.

As we aim at high tagging precision, the verb guesser is semi-automatic in the sense of automatically providing analyses for clear-cut cases, while leaving cases of ambiguity and data sparseness to manual judgement. In this sense, the guesser is ambiguity aware, and unlike a statistical tool, it provides categorical results (noun/verb/other) for the large majority of candidates.

As the verb guesser makes use of linguistic knowledge about verbal suffix clusters, an obvious secondary use of this device, planned for the future, is to provide a detailed morphological analysis of the verb forms which it identifies. This could in turn serve as an input to syntactic and semantic processing of Northern Sotho.

Ackowledgements — Financial support from the National Research Foundation (NRF) for this project is gratefully acknowledged.

References

De Schryver G-M & De Pauw G. 2007. Dictionary Writing Systems (DWS) + Corpus Query Package (CQP): The case of TshwaneLex. *Lexikos **17***: 226–246.

De Schryver G-M & Prinsloo DJ. 2000. The compilation of electronic corpora, with special reference to the African languages. *Southern African Linguistics and Applied Language Studies* **18**(1-4): 89–106.

Heid U, Prinsloo DJ, Faaß F, & Taljard E. Forthcoming. Designing a noun guesser for part of speech tagging in Northern Sotho.

Kotzé PM & Anderson WN. 2006. Finite state tokenisation of an orthographical disjunctive agglutinative language: The verbal segment of Northern Sotho. Proceedings of the 5th International Conference on Language Resources and Evaluation, Genoa, Italy, pp 1906–1911.

Lombard DP. 1985. *Introduction to the Grammar of Northern Sotho*. Pretoria: J.L. van Schaik Publishers.

Louwrens LJ. 1991. *Aspects of Northern Sotho Grammar.* Pretoria: Nasou Via Afrika.

Poulos G & Louwrens LJ. 1994. *A Linguistic Analysis of Northern Sotho*. Pretoria: Via Afrika.

Prinsloo DJ & Heid U. 2006. Creating word class tagged corpora for Northern Sotho by linguistically informed bootstrapping. In Ties I (ed.) LULCL – Lesser used languages and computer linguistics. Proceedings of the Conference for Lesser Used Languages and Computer Linguistics, EURAC research, European Academy, 27–28 October 2005, Bolzano, Italy, pp 97–115.

Thobakgale RM. 'Khuetšo ya OK Matsepe go bangwadi ba Sepedi' (Doctoral thesis, University of Pretoria, 2005).

Van Wyk EB, Groenewald PS, Prinsloo DJ, Kock JHM & Taljard E. 1992. *Northern Sotho for first-years*. Pretoria: J.L van Schaik Publishers.

Ziervogel D & Mokgokong PD. 1975. *Groot Noord-Sotho woordeboek*. Pretoria: J.L. van Schaik Publishers.

Appendix 1: Three hundred randomly selected verbs from *GNSW*

aba (1), aba (2), abela, abula, adibetša, adima, aena, aga, ahlaahla, ahlama, ahloga, ahlola, aka, akgofa, alafa, bala, bofa, boifa, bolaya, boloka, bona, bopa, dira, diša, dita, ditela, duba, dubaduba, duduetša, dukola, dukologa, dula, duma (1), duma (2), dumaduma, dumaela, dumala, dupa, duša, duta, dutla, dutlalala, dutula, ebaeba, ebela, eboga, ebola, edimola, efa, ega, ehula, eka (1), eka (2), eka (3), ekelela, eketša, ekiša, ela (1), ela (2), ela (3), eleletša, elelwa, ema, emaema, emara, ena (1), ena (2), enega, enela, enta, enya, epa (1), epa (2), epela, eta, ethimola, etiša, etša (1), etša (2), etsela, fa, fadimega, fafama, fafanyega, fafatla, fafatša, faga, fagafagetša, fagahla, fagola, fahla (1), fahla (2), fahloga, fahlosela, faka, fakaša, fakasela, fakatšha, fala (1), fala (2), falala, falodiša, falola, famoga, famola, fantsatsa, fantšha, fanyetša, faola, faolla, faologa, fapa, fapafapana, fapoga, fapologa, fara, farafara, faragetša, farakana, faraša, fasa, fasola, fata, fatagana, fatalala, fatankola, fatoga, fatola, fatša, fea, feafea, feama, feana, feba, fefa (1), fefa (2), fefeana, fefera, fefoga, fefola, fefotša, fega, fegafega, fegelela, fegeletša, fegelwa, fehla, feila, feka, fekeetša, fekemela, fekola, fela (1), fela (2), felega, felegetša, feleka, felesetša, fema (1), fema (2), fema (3), fena, fenama, fenoga, fenola, fenya, fenyeka, fenyekula, fepa, fera, ferea, ferefa, ferefehla, ferehla, ferekana, ferekega, ferelela, feroga, ferola, feta, fethekga, fetla, fetoga, fetola, fifala, fihla (1), fihla (2), fihlola, fina, finilala, fiša, fiša (1), fiusa, foa, fofa, fofora, fofothela, fofotša, fogohla, fogohlela, fohla, foka, fokafoka, fokoga, fokola, fokotša, fola (1), fola (2), fola (3), fola (4), folagana, follela, folletša, fologa, folotša, foma, fonkodiša, fonkotša, fopha, fora (1), fora (2), foraforetša, forela, forohla, foroma, foša, fotha, fotla (1), fotla (2), foufala, gabea, gadia, gadika, gafa, gagatla, gahla, gaketla, galaka, galefa, ganka, gatla, gega, gegea, gitla, gobotla, gobua, gohla, gokga, gokotla, golega, gomea, gonoka, gotla, gwanta, gwata, hlalefa, hlehla, hleka, hloka, hlweka, hupa, iketla, kaba, kabata, kadiela, kaela, kaka, kakabala, kakabolla, kakadia, kakaila, kakamala, kata, kata, kata, kgafa, koba, lahla, laiša, laka, lama, lapa, lata, latswa, leba, lefa, natefa, natla, nea, neka, ngwala, paka, paka, pitika, rafa, rata, reka, šonya, taba, taboga, tadima, taela, taga, tagafala, taila, taka, thuša, tšwafa

Appendix 2: List of suffixes and combinations of suffixes (suffix clusters)
Ambiguous suffix (clusters) are marked as bold italic

-a, -aditše, -aditšwe, -afaditše, -afaditšwe, -afala, -afaleditše, -afaleditšwe, -afaletša, -afaletšana, -afaletšane, -afaletšanwa, -afaletšanwe, -afaletšwa, -afalwa, -afatša, -afatšwa, -afetše, -afetšwe, -agaditše, -agaditšwe, -agala, -agaleditše, -agaleditšwe, -agalela, -agalelwa, -agaletša, -agaletšana, -agaletšane, -agaletšanwa, -agaletšanwe, -agaletše, -agaletšwa, -agaletšwe, -agalwa, -agana, -agane, -agantše, -agantšha, -agantšhitše, -agantšhitšwe, -agantšhwa, -agantšwe, -aganwa, -aganwe, -aganya, -aganywa, -agatša, -agatšwa, -agetše, -agetšwe, ***-aka***, -akana, -akane, -akantše, -akantšha, -akantšhitše, -akantšhitšwe, -akantšhwa, -akantšwe, -akanwa, -akanwe, -akanya, -akanywa, ***-akile***, -akilwe, -akwa, ***-ala***, -aladitše, -aladitšwe, -alala, -alalela, -alalelana, -alalelane, -alalelanwa, -alalelanwe, -alalelwa, -alaletše, -alaletšwe, -alalwa, -alatša, -alatšeditše, -alatšeditšwe, -alatšetša, -alatšetšwa, -alatšwa, -aletše, -aletšwe, -alwa, -ama, -ame, -amela, -amelwa, -ametše, -ametšwe, -amiša, -amišana, -amišane, -amišanwa, -amišanwe, -amišitše, -amišitšwe, -amišwa, ***-ana***, ***-ane***, -anela, -anelwa, -anetše, -anetšwe, ***-ang***, -angwa, -angwe, -antše, -antšha, -antšheditše, -antšheditšwe, -antšhetša, -antšhetšana, -antšhetšane, -antšhetšanwa, -antšhetšanwe, -antšhetšwa, -antšhitše, -antšhitšwe, -antšhwa, -antšwe, -anwa, -anwe, -anya, -anyeditše, -anyeditšwe, -anyetša, -anyetšana, -anyetšane, -anyetšanwa, -anyetšanwe, -anyetšwa, -anywa, -ara, -areditše, -areditšwe, -arega, -aregile, -arela, -arelela, -arelelana, -arelelane, -arelelanwa, -arelelanwe, -arelelwa, -areletše, -areletšwe, -arelwa, -aretša, -aretšana, -aretšane, -aretšanwa, -aretšanwe, -aretše, -aretšwa, -aretšwe, -ariša, -arišana, -arišane, -arišanwa, -arišanwe, -arišitše, -arišitšwe, -arišwa, -arolla, -arollela, -arollelana, -arollelane, -arollelanwa, -arollelanwe, -arollelwa, -arolletše, -arolletšwe, -arolliša, -arollišana, -arollišane, -arollišanwa, -arollišanwe, -arollišitše, -arollišitšwe, -arollišwa, -arollwa, -arologa, -arologile, -arologilwe, -arologwa, -arolotše, -arolotšwe, -arrollana, -arrollane, -arrollanwa, -arrollanwe, -arwa, -atša, -atšega, -atšegile, -atšegilwe, -atšegwa, -atšwa, -bja, -dile, -dilwe, -ditše, -ditšwe, ***-e***, -ediša, -edišana, -edišane, -edišanwa, -edišanwe, -edišitše, -edišitšwe, -edišwa, -editše, -editšwe, ***-ega***, -egela, -egelana, -egelane, -egelanwa, -egelanwe, -egelwa, -egetše, -egetšwe, -egile, ***-ela***, -elana, -elane, -elanwa, -elanwe, -elediša, -eledišana, -eledišane, -eledišanwa, -eledišanwe, -eledišitše, -eledišitšwe, -elediša, -eleditše, -eleditšwe, -elega, -elegile, -elegilwe, -elegwa, -elela, -elelana, -elelane, -elelanwa, -elelanwe, -elelega, -elelegile, -elelegilwe, -elelegwa, -elelwa, -eletša, -eletšana, -eletšane, -eletšanwa, -eletšanwe, -eletše, -eletšwa, -eletšwe, -elwa, ***-eng***, -ere, -erwe, ***-etša***, -etšana, -etšane, -etšanwa, -etšanwe, -etše, ***-etšwa***, -etšwe, -fša, -ile, -ilwe, -iša, -išana, -išane, -išanwa, -išanwe, -išeditše, -išeditšwe, -išega, -išegile, -išegilwe, -išegwa, -išetša, -išetšana, -išetšane, -išetšanwa, -išetšanwe, -išetšwa, -išiša, -išišana, -išišane, -išišanwa, -išišanwe, -išišitše, -išišitšwe, -išišwa, -išitše, -išitšwe, -išwa, -itše, -itšwe, -iwa, -me, ***-ne***, -ngwa, ***-nne***, -nnwe, -oga, -ogana, -ogane, -oganwa, -oganwe, -ogela, -ogelwa, -ogetše, -ogetšwe, -ogile, -ogilwe, -ogwa, ***-ola***, -olana, -olane, -olanwa, -olanwe, -olega, -olegile, -olegilwe, -olegwa, -olela, -olelana, -olelane, -olelanwa, -olelanwe, -olelwa, -oletše, -oletšwe, -olla, -ollana, -ollane, -ollanwa, -ollanwe, -ollega, -ollegile, -ollegilwe, -ollegwa, -ollela, -ollelana, -ollelane, -ollelanwa, -ollelanwe, -ollelwa, -olletše, -olletšwe, -olliša, -ollišana, -ollišane, -ollišanwa, -ollišanwe, -ollišitše, -ollišitšwe, -ollišwa, -ollwa, -ologa, -ologana, -ologane, -ologanwa, -ologanwe, -ologile, -ologilwe, -ologwa, -ološa, -ološana, -ološane, -ološanwa, -ološanwe, -ološeditše, -ološeditšwe, -ološetša, -ološetšana, -ološetšane, -ološetšanwa, -ološetšanwe, -ološetšwa, -ološitše, -ološitšwe, -ološa, -olotše, -olotšwe, -olwa, -oša, -ošeditše, -ošeditšwe, -ošetša, -ošetšana, -ošetšane, -ošetšanwa, -ošetšanwe, -ošetšwa, -ošitše, -ošitšwe, -ošwa, -otše, -otšwe, -pša, -pšha, ***-tše***, -tšwe, ***-wa***, ***-we***