Towards Second-Generation Spellcheckers for the South African Languages

D.J. PRINSLOO[°] & Gilles-Maurice DE SCHRYVER[‡]
Department of African Languages, University of Pretoria, SA[°‡] &
Department of African Languages and Cultures, Ghent University, Belgium[‡]

Abstract: In this paper we present spellcheckers for the South African languages, viz. for the nine official African languages and for Afrikaans (English already being catered for in the group of world-English spellcheckers). The first section is devoted to (i) describing certain basic aspects regarding the functionalities of spellcheckers, and to (ii) some specific African-language issues. This is followed by a brief evaluation of spellcheckers currently available for Afrikaans and some of the African languages. The final part deals with more advanced principles underlying spellcheckers with a view to create the next generation of spellcheckers for the South African languages.

1. Human Language Technology (HLT) and spellcheckers

From the early 1960s onwards, researchers have designed various methods for the automatic detection of erroneous words in running text. Today, four decades later, there isn't any self-respecting word processor that doesn't include a spelling checker, as well as a spelling suggestor and/or corrector, a grammar checker and even a thesaurus as an integral part. This is true for all languages with significant worldwide commercial importance, less so for those languages with a limited commercial value. When we focus on the African languages[1], we must sadly note that commercially available spellcheckers are unfortunately the exception rather than the rule. It can hardly be disputed that the use and development of spellcheckers for the African languages at large are still in their infancy. For most African languages no spellcheckers exist and for those languages for which spellcheckers are available, the actual use is questionable.

All efforts regarding state-of-the-art, high-tech development of especially the African languages should be applauded. We believe however that such activities and the development strategies should be sensitive to certain realities of the South African situation and should address Human Language Technology (HLT) needs on a *priority basis* rather than on an *ideal*-HLT-development schedule. This means that major projects should be designed in such a way as to render *regular spin-offs*, i.e. usable products that are urgently needed. This might even entail taking shortcuts in the short term in order to provide products for immediate use for which the technology is in real

[1] Since this paper is being submitted for publication in South Africa, necessary sensitivity with regard to the term 'Bantu' languages is exercised in the authors' choice rather to use the term *African* languages. Keep in mind, however, that the latter includes more than just the 'Bantu Language Family'.

terms still under development. African languages in particular need what we call first-generation spellcheckers *now* to satisfy the immediate needs which could be described as spellcheckers that can detect most incorrectly typed words and suggest alternatives. This should be followed by subsequent, more sophisticated and improved spellcheckers which can also check grammatical structures. We thus believe that if ways can be found to satisfy the immediate needs of the users of specific languages, the process should not be delayed simply for the sake of releasing a more sophisticated spellchecker as the first product.

2. Brief theoretical conspectus on spellcheckers

The term 'spellchecker' is used here to cover what the average user understands under this term today, i.e. a piece of software, mostly integrated into a word processor like Microsoft Word or Corel WordPerfect, which (i) *checks* for spelling (and grammatical) errors, (ii) *automatically corrects* some typos, (iii) makes *suggestions* for other mistakes, and (iv) often includes a *thesaurus* (i.e. a list with synonyms and antonyms).

Viewed from the angle of the compiler of a spellchecker, Kukich (1992), still one of the definitive reference works, points out that three types of distinctions must be made: (i) error *detection* versus error *correction*; (ii) *interactive* spelling checkers versus *automatic* correction; and (iii) attention to *isolated* words versus linguistic or textual *context*. These distinctions result in the fact that research in this field *'has focused on three progressively more difficult problems: (1) non-word error detection; (2) isolated-word error correction; and (3) context-dependent word correction'* (Kukich 1992: 377).

Basically there are two main approaches to spellcheckers. Firstly, one can program software with a proper description of a language, including detailed morphophonological and syntactic rules, which computes over a stored list of word-roots. Secondly, one can simply compare the spelling of typed (or scanned) words with a stored list of top-frequency orthographic word-forms.

3. Issues in the design of spellcheckers for the South African languages

As far as we know, the only commercially available spellcheckers for the African languages are the one developed for Kiswahili by Arvi Hurskainen (Microsoft Word; cf. Hurskainen 1999: 139), and the first-generation series for isiZulu, isiXhosa, Sepedi and Setswana developed by D.J. Prinsloo (Corel WordPerfect; cf. Prinsloo & De Schryver 2001: 129). Spellcheckers known to the authors for Afrikaans are the commercially available products by Corel WordPerfect, Pharos, and the University of Potchefstroom for CHE.

In oversimplified terms it can be said that the purpose of a spellchecker in word processing software is to alert the user to possibly incorrectly-typed words or strings and to suggest options for correction. It can of course be argued that the principles underlying error detection and the techniques to suggest improvements are languageindependent. There are however certain unique characteristics of African languages that require adjustments in the approach to e.g. error detection. A good example in this regard is the handling of occurrences of sequences of equal words. One of the typical errors made in text production in any language is indeed the erroneous repetition of a word (*the the* is common in English). Therefore, a standard error-detecting function in spellcheckers is to highlight occurrences of supposedly-erroneous sequences of equal words. For the disjunctively-written African languages this, unfortunately, results in the highlighting of a huge number of *correctly typed double, triple, ... words*. For these languages this function is counterproductive because it delays the process of verification of correctness rather than contributing to it. Secondly, the handling of special characters in spellcheckers for African languages is a problematic issue. Ideally, provision should be made for all 'special characters' (i.e. those with a Latin base cum diacritics) used in these languages such as *š* and *Š* in Sepedi, and a fairly extensive number for Tshivenda, just as is the case for special characters like *ø* in Danish or *ç* in French. The Sepedi *š* and *Š* pose no problem for either compiler or user of spellcheckers since these characters have been assigned standard ASCII values, namely 0154 for *š* and 0138 for *Š*. Both programmer and user can therefore easily create them. This, however, is not the case in Tshivenda where the average user does not have a special character set on his/her computer. Moreover, albeit words typed without the diacritics could even be (semi-)automatically converted to the correct orthography by a Tshivenda spellchecker, such texts will create problems in printouts, e-mail correspondence and Internet up- or downloads and this will in the end be counterproductive unless certain specific solutions could be found.

4. A brief evaluation of currently available spellcheckers for the South African languages

In this section answers are sought to the questions:

- Is it possible to obtain acceptable error-detection levels for South African languages using spellcheckers solely based on top-frequency wordlists?
- What does the average user regard as a minimum or satisfactory level of success?
- Will the success rate be comparable for conjunctively and disjunctively written languages? Or thus, should a different approach be followed for the Nguni languages (isiZulu, isiXhosa, siSwati and isiNdebele) on the one hand, and the Sotho languages (Sepedi, Sesotho, Setswana) as well as Tshivenda and Xitsonga on the other?

A statistical evaluation – part of a much larger study – of the situation for Afrikaans, and then for isiZulu and Sepedi will now be attempted.

For Afrikaans the effectiveness of the three commercial spellcheckers, viz. Corel WordPerfect, Pharos, and Potchefstroom, was tested on a variety of randomly selected texts. For the purpose of this paper only a brief summary of the outcome will be offered, exemplified on a small section from the *White Pages*, as shown in Table 1.

Table 1: Spellchecking a randomly selected Afrikaans section from the *White Pages* (1999-2000: 14)

Afrikaans spellchecker	**Number of words in sample**	**Number of correct words *not* recognised**	**Success rate**
Corel WordPerfect	203	11	94.6%
Pharos	203	7	96.6%
Potchefstroom	203	11	94.6%

From Table 1 it is clear that the overall percentage of error detection is quite acceptable. From the subsequent experiments, however, it became clear that all three spellcheckers do not fare well with the numerous *compounds* characteristic of the Afrikaans language, a problematic situation from a users' point of view. This thus reflects the 'limits' of first-generation spellcheckers for Afrikaans based on top-frequency wordlists.

Turning to the African languages, tests were conducted on two randomly selected paragraphs, (1) and (2) below. A single glance at these texts immediately reveals that isiZulu has a conjunctive orthography while Sepedi is written disjunctively. In (1) the isiZulu paragraph is shown, where the word-forms in bold are not recognised by the Corel WordPerfect spellchecker software.

(1) Spellchecking a randomly selected Zulu paragraph from *Bona Zulu* (June 2000: 114)

Izingane **ezizichamelayo** zivame ukuhlala **ngokuhlukumezeka** kanti akufanele **ziphathwe** ngaleyondlela. Uma ushaya ingane ngoba **izichamelile** usuke **uyihlukumeza** ngoba lokho **ayikwenzi** ngamabomu njengoba iningi labazali **licabanga** kanjalo. Uma nawe **mzali usubuyisa** ingqondo, usho ukuthi ikhona ingane **engajatshuliswa wukuvuka** embhedeni obandayo **omanzi** njalo ekuseni?

The stored isiZulu list consists of the 33,526 most frequently used word-forms. As 12 out of 41 word-forms were not recognised in (1), this implies a success rate of 'only' 70.7%.

When we test the Corel WordPerfect spellchecker software on a randomly selected Sepedi paragraph, however, the results are as shown in (2).

(2) Spellchecking a randomly selected Sepedi section from the *White Pages* (1999-2000: 24)

Dikarata tša mogala di a hwetšagala ka go fapafapana goba R15, R20, (R2 ke mahala) R50, R100 goba R200. Gomme di ka šomišwa go megala ya **Telkom** ka moka (ye metala) Ge tšhelete ka moka e fedile **karateng** o ka tsentšha karata ye nngwe ntle le go šitiša poledišano ya gago mogaleng.

Even though the stored Sepedi list is smaller than the isiZulu one, as it only consists of the 27,020 most frequently used word-forms, with 2 unrecognised words out of 46, the success rate is as high as 95.7%.

In an extensive second series of experiments the aim was to establish the error-detection power resulting from the cumulative build-up of top-frequency wordlists as the basis for spellcheckers for these languages. It was found that Sepedi reaches an acceptable success rate with a much smaller word list than for Afrikaans and that the success rate for isiZulu is lower even when very large word lists are used.

From a users' perspective, the success rate of a first-generation spellchecker for a conjunctively-written language like isiZulu is not really acceptable. Disjunctivism is however a great advantage for isolated-word spellchecking, as is clear from the Sepedi data. For Afrikaans, large wordlists can 'just' do.

5. Towards second-generation spellcheckers

From the above it follows that advanced technologies, as for English, for example, should be developed in what we prefer to call second-generation spellcheckers for Afrikaans, to cater for compounds in another way than the mere stacking of words in a wordlist. First-generation spellcheckers for Afrikaans could thus be improved by programming sets of rules for compounding. A spellchecker based on a true morphological analyser / generator of the language is, however, the ideal solution.

For the African languages it is clear that, for isolated-word spellchecking purposes of the Nguni languages, second-generation spellcheckers are needed to reach a more satisfactory rate of error detection. With this in mind, a thorough study was undertaken of the degree of conjunctivism / disjunctivism of all official South African languages. The results of this endeavour are shown in Table 2.

Table 2: Degrees of conjunctivism / disjunctivism for the South African languages (based on counts derived from 55 two-by-two parallel corpora, cf. Prinsloo & De Schryver 2002: 261)

	isiNdebele	Siswati	isiXhosa	isiZulu	English	Afrikaans	Xitsonga	Setswana	Tshivenda	Sepedi	Sesotho
isiNdebele	**1.00**	1.01	1.01	1.04	1.41	1.41	1.61	1.63	1.67	1.73	1.77
Siswati	0.99	**1.00**	1.03	1.04	1.41	1.41	1.61	1.62	1.69	1.72	1.77
isiXhosa	0.99	0.97	**1.00**	1.01	1.36	1.37	1.58	1.58	1.75	1.67	1.71
isiZulu	0.96	0.97	0.99	**1.00**	1.32	1.34	1.54	1.55	1.58	1.60	1.66
English	0.71	0.71	0.74	0.76	**1.00**	1.00	1.15	1.16	1.19	1.24	1.25
Afrikaans	0.71	0.71	0.73	0.75	1.00	**1.00**	1.15	1.16	1.19	1.23	1.24
Xitsonga	0.62	0.62	0.63	0.65	0.87	0.87	**1.00**	1.01	1.05	1.06	1.08
Setswana	0.62	0.62	0.63	0.64	0.86	0.86	0.99	**1.00**	1.03	1.07	1.08
Tshivenda	0.60	0.59	0.57	0.63	0.84	0.84	0.96	0.97	**1.00**	1.03	1.08
Sepedi	0.58	0.58	0.60	0.62	0.81	0.81	0.94	0.94	0.97	**1.00**	1.02
Sesotho	0.57	0.57	0.58	0.60	0.80	0.80	0.92	0.92	0.93	0.98	**1.00**

From Table 2 one can for instance see that Sepedi is 60% more disjunctive than isiZulu, or that isiNdebele is 57% more conjunctive than Sesotho. The figures in this table have a *direct* impact on the success rate of spellcheckers for the African languages, as a higher degree of conjunctivism implies a lower degree of success rate.

In contrast to Afrikaans, the main error-detection problem for the African languages is not one of compounding, but one of morphophonological changes resulting from the agglutination of morphemes in especially the Nguni languages. It is thus suggested that a proper morphological analyser / generator be incorporated into the second-generation spellcheckers for the African languages – and finite-state tools are indeed already being developed to this end, cf. e.g. Bosch & Pretorius (2002) for isiZulu, or De Schryver (2002b) for Sepedi.

Looking ahead, to the third-generation spellcheckers, these will of course need to have a grammar component as well. For the disjunctively-written languages this will for instance 'solve' the current problem that a correct sequence of two or more equal words is marked as potentially wrong.

6. Conclusion

We have seen that first-generation spellcheckers, viz. spellcheckers based on top-frequency wordlists, result in *just acceptable* error-detection software for a language like Afrikaans which is characterised by extensive compounding. Conversely, this same approach produces *excellent* error-detection software for disjunctively-written South African languages. For the conjunctively-written South African languages (the Nguni languages), however, even long lists of word-forms can *not really* be considered *acceptable*. The success of isolated-word error detection for the African languages is inversely related to the degree of conjunctivism.

It was further suggested that the second generation of spellcheckers for Afrikaans include some basic compounding rules, and that the conjunctively-written South African languages include a morphological analyser / generator. The latter will of course be a crucial component of *all* South African third-generation spellcheckers – spellcheckers which will also be able to perform grammatical checks.

References

Bona Zulu, Imagazini Yesizwe, Durban, June 2000.

Bosch, S.E. and L. Pretorius. 2002. Using Finite-State Computational Morphology to Enhance a Machine-Readable Lexicon. In G.-M. de Schryver (ed.). 2002a: 20-22.

De Schryver, G.-M. (ed.). 2002a. *AFRILEX 2002, Culture and Dictionaries, Programme and Abstracts*. Pretoria: $(SF)^2$ Press.

De Schryver, G.-M. 2002b. First Steps in the Finite-State Morphological Analysis of Northern Sotho. In G.-M. de Schryver (ed.). 2002a: 22-23.

Hurskainen, A. 1999. SALAMA: Swahili Language Manager. *Nordic Journal of African Studies* 8/2: 139-157.

Kukich, K. 1992. Techniques for Automatically Correcting Words in Text. *ACM Computing Surveys* 24/4: 377-439.

Prinsloo, D.J. and G.-M. de Schryver. 2001. Corpus applications for the African languages, with special reference to research, teaching, learning and software. *Southern African Linguistics and Applied Language Studies* 19/1-2: 111-131.

Prinsloo, D.J. and G.-M. de Schryver. 2002. Towards an 11 x 11 Array for the Degree of Conjunctivism / Disjunctivism of the South African Languages. *Nordic Journal of African Studies* 11/2: 249-265.

White Pages Pretoria, North Sotho – English – Afrikaans Information Pages, Johannesburg, 1999-2000.