Syllable analysis of North Sotho and its effect on computerized hyphenation

C.S. Price and Q.H. Gee

The need to address multiple languages in word processing is becoming increasingly important. Most word processors have some hyphenation facility which will split words occurring at the end of a line into two subwords. However, most word processors' hyphenation facilities cater only for American English, while some cater for British English, and a few for South African English. This means that if one is to process text written in a language other than English, the automatic hyphenation will produce incorrect and maybe even nonsensical subwords. In this case, the hyphenation facility will have to be disabled.

Some work has been performed on the hyphenation of words: Ocker's work is one of the many done on the English language.[1] Even though English has many exceptions, an algorithm has been developed to find all possible hyphenation points without much semantic and syntactic knowledge. Mañas[2] has formulated an algorithm for the hyphenation (and syllabication) of Spanish words: this algorithm needs only morphological knowledge to determine the hyphenation points. Other work done for languages spoken in South Africa include the hyphenation of Afrikaans by Gee[3] and Zulu by Hampton.[4]

This article presents an algorithm for the hyphenation and syllabication of North Sotho words. It does not require any syntactic or semantic knowledge of the word to be hyphenated. The hyphenation and syllabication of the words are very similar, and their differences are described. The algorithm finds all possible hyphenation points of a word, but does not choose a particular hyphenation point. This will be done by the calling procedure, which is probably part of the word processor package. The results from an implementation of the algorithm are also presented.

Introduction to North Sotho

North Sotho (Sepedi or Pedi) is a language which is spoken predominantly in the northern Transvaal. The work of transcribing the language took place between the mid and late 19th century, with the first reader appearing in 1870,[5] the first excerpts of the New Testament in 1890 and the complete Bible in 1904.[6] The language is written with Roman characters and has undergone three major changes to its orthography (the way the language is written). These three changes are marked by the periods 1870 – 1951, 1951 – 1985 and 1985 to the present day, respectively. Most published material in schools and bookshops are typeset using the second orthography. This orthography is therefore the one considered here. It is conjectured that the use of other orthographies does not pose problems for the hyphenation and syllabication algorithms presented.

The orthography chosen uses the same characters to represent vowels and consonants as English does. It has an additional consonant, viz. *š*, which is pronounced as the English 'sh', but the diacritic mark makes no difference to the phonetic value of the word when considering it for hyphenation. In literary texts, the two vowels *â* and *ê* can be found[7] in addition, but the circumflex does not change the phonetic value of the word, and the accent can therefore be omitted for our purposes.

Because North Sotho was transcribed recently (in comparison to English), a closer mapping has been achieved between the syllables, sounds and morphological components, and the orthographic characters which represent them. This makes the rules for hyphenation and syllabication much simpler than the rules for English. Rules for the hyphenation of Spanish words are simpler than those of English,[2] but they are more complex than those for North Sotho.

Rules for hyphenation and syllabication

By hyphenation we mean the breaking of written words so that the resulting subwords consist of complete syllables. The syllable refers to acceptable sound sequences of consonants and vowels, together with features such as length and stress, or to single consonants or vowels, which are suitably considered as a group for analysis. For our purposes syllabication means the decomposition of words into these syllables. Thus the difference is that hyphenated subwords always have at least one written vowel per syllable, whereas with syllabication the minimum component can be a single vowel or a consonant cluster.

North Sotho hyphenation and syllabication are identical except where certain consonant combinations occur at the beginning or at the end of a word. In what follows below, 'hyphenation' means both hyphenation and syllabication, unless otherwise stated.

Consonant clusters

A consonant cluster consists of one or more consonants. All consonants forming such a cluster are pronounced as one sound. Hence, even though there are two, three or four consonants consecutively in a word, they can never be separated by hyphens. A table showing some of the most common consonant clusters with examples of words containing these clusters is shown below. Clusters with only one consonant are not shown.

Consonant cluster	*Example of word*
bj	bjalo, bja, bjo
fs	bofsa, kgaufsi
fš	fša
gw	gagwe
hl	Moahlodi, kahlolo, mehla
hw	hwetše
kg	Mokgethwa, kgopo, kgalefo
kgw	nkgwana, mokgwa
kh	khunollo, khorišago, khutsong
kr	disakramente
kw	kwang, kwa, kwena
ll	khunollo
lw	ngwadilwego, bolelwa
mm	mma, mme
nd	Maikondijo
ng	kerekeng, melaong, seng
ngw	ngwadilwego, ngwana
nw	nwa
ny	Bosenyi, tshwenyegilego, nyatše
nyw	tsenywang, enywago
ph	diphetogo, phele, mphetoša, bophelong
pš	eupše
psh	mphafatša, mpshfatšwa
py	Mpye
rw	rwala, feberware
sw	swanetse
šw	tlišwe
th	mpetha, motho
tl	tlišwe, tlogela, tla, setlegilego
tlw	mebitlwa
ts	tsenywang, tseba
tsh	boitsholo
tshw	tshwenyegilego
tsw	tswalela
tš	nyatše, tša, beetšego, letšatši
tšh	ntšhitswe
tšw	mpshafatšwa

The treatment of vowels

The hyphenation and syllabication of North Sotho words revolve mainly around the vowels. When a vowel cluster is followed by a consonant cluster, a hyphen is always inserted between them. However, an exception may occur in words ending in *ng*, depending on whether hyphenation or syllabication is required. The *ng* at the end of a word is pronounced as a separate syllable, so if syllabication is required, the above rule does not need to be modified. If hyphenation is required, there should be no hyphenation point between the root of the word and the *ng* as it is not considered

The authors are in the Department of Computer Science, University of the Witwatersrand, P.O. Wits, Johannesburg 2050, South Africa.

good practice to allow such a word to be hyphenated before the *ng*.[8,9] This is the first differentiation made between hyphenation and syllabication.

Prefixes and suffixes

North Sotho is a language which does not have multiple prefixes and suffixes. It is useful to recognise and remove the suffixes and prefixes (affixes) from the root of the word when hyphenating for two reasons: first, when the affix is attached to the root of the word, the hyphenation rules incorrectly perform the hyphenation. Secondly, there is always a hyphenation point between the affixes and the root of the word, so the affix and the root can be considered to be different entities to be hyphenated.

Prefixes

Most often, prefixes in North Sotho indicate singular or plural words. Other types of prefix exist, which could be classed as part of a consonant cluster, except for the fact that they are pronounced as two syllables. These are

- *m* followed by *p*, e.g. m-psha-fa-tsa
- *n* followed by *k, n* or *t*, e.g. n-te-ka

However, it is not considered good practice to separate the *m* or *n* from the rest of the word in hyphenating, even though it is syllabically correct.[8,9] If syllabication is required, a hyphen is inserted after the *m* or *n*. The prefixes are shown in the table below. Our notation is that __ represents a blank space, i.e. the start of a word, and that - represents a valid hyphenation point. Upper and lower cases are not significant.

Prefixes	*Followed by*	*Hyphenation*	*Syllabication*
(Singular/plural prefixes)			
__bo		__bo-	__bo-
__le		__le-	__le-
__mo		__mo-	__mo-
__se		__se-	__se-
__ba		__ba-	__ba-
__di		__di-	__di-
__ma		__ma-	__ma-
(Other prefixes)			
__m	p	__mp	__m-p
__n	k	__nk	__n-k
__n	n	__nn	__n-n
__n	t	__nt	__n-t

Suffixes

There are several suffixes in North Sotho which mostly denote the tense of the verb. An extension to the rule of hyphenating before the suffix *-ng* is: if two vowels precede the *ng*, then a hyphen is placed between them. The common suffixes are shown in the table below. Note that __ again represents the blank space following the end of a word. Upper and lower cases are not significant. V denotes any vowel.

Suffixes	*Hyphenation*	*Syllabication*
ela__	-e-la__	-e-la__
etšego__	-e-tše-go__	e-tše-go__
etšo__	-e-tšo__	-e-tšo__
ilwego__	-il-we-go__	-il-we-go__
ng__	-ng__	-ng__
olola__	-o-lo-la__	-o-lo-la__
wa__	-wa__	-wa__
VVng__	V-Vng__	V-V-ng__

Double vowels

If there are two consecutive vowels in a word, and if the above rules have not dealt with them, then two further cases must be tested. If these vowel pairs are *ai, ee, ei, eu, ii* or *oo*, they should not be separated by a hyphen. If they do not fall under this category, they should be separated. A list of vowel clusters and vowel pairs is shown below.

Non-separable vowel pair	*Example of word*
ai	seswai, seswaing, letšwai
ee	itšeela, gotee, meetse
ei	eitše, leineng
eu	eupše
ii	Tiišetšo
oo	diphoofolo
Separable vowel pair	*Example of word*
ae	taelo, Isiraele
ao	molao, maoto, laola
au	Dikgaugelo
ea	seatleng
eo	yeo, Lenaneo
ia	batiakone
oa	Moahlodi
oi	koloi

Figure 1 summarises these rules. Note that syllabication is an extended form of hyphenation: it differs only in Rules 2a, 7 and 8.

Exceptions

Two exceptions to the above rules have been found so far. They are:

1) *lefeela*: here the *ee* should remain unseparated as *ela* here is not a suffix. *le-fee-la.*

2) *maatla*: here the *ma* does not denote a plural word; hence the *aa* should not be separated. *maa-tla.*

There may be more exceptions, but they have not yet been identified.

Implementing the rules

A practical algorithm

In the hyphenation or syllabication of North Sotho words, the vowels and their positions in the word relative to other vowels or consonants are significant. Moreover, there are more instances where hyphens should be inserted than not. Thus insertion of hyphens followed by the removal of unneeded ones is an easier process than checking where hyphens should be inserted.

The algorithm shown in Fig. 2 finds all the possible hyphenation (or syllabication) points of a given word and inserts a hyphen at each of those points. It can easily be modified to save the position of the hyphenation (or syllabication) points (e.g. hyphenate after the 3rd, 7th or 9th letter) for subsequent printing. The algorithm works in two phases on one word: first by inserting hyphens, and secondly by removing unnecessary ones.

In this algorithm, only the suffixes beginning with a vowel need to be stored. This saves space, as suffixes beginning with consonants are automatically dealt with by Rule No. 2. Rule No. 4 is automatically dealt with by step 3.1(1) of the algorithm: hyphens are inserted between the vowels. Neither vowel pairs to be separated, nor consonant clusters, need to be stored.

Environment

The machine used to implement the program was an IBM 3083 J24 mainframe running the OS/CMS operating system. The language used to implement the program was Pascal and the Pascal/VS compiler was used. This program can be transferred to and implemented on an IBM PC if required.

Results

Tables showing some sample output are shown below. The program which imple-

Rule No.	*Hyphenation*	*Syllabication*
1	CC→CC	CC→CC
2a	Vng__→Vng__	Vng__→V-ng__
2	VC→V-C	VC→V-C
3	VS→V-S	VS→V-S
4	VV*ng*__→V-V*ng*__	VV*ng*__→V-V-*ng*__
5	P(n)V→P(n)-V	P(n)V→P(n)-V
6	P(n)C→P(n)-C	P(n)C→P(n)-C
7	__mC_p→__mC_p	__mC_p→__m-C_p
8	__$nC_{k/n/t}$→__$nC_{k/n/t}$	__$nC_{k/n/t}$→__n-$C_{k/n/t}$
9	V(vp)V(vp)→V(vp)V(vp)	V(vp)V(vp)→V(vp)V(vp)
10	VV→V-V	VV→V-V

Where

__is a blank space
V is a vowel
C is a consonant cluster
S is a suffix
P(n) is a prefix of the singular/plural type
C_p is a consonant cluster beginning with *p*
$C_{k/n/t}$ is a consonant cluster beginning with *k*, n or *t*
V(vp) is one of the vowels in an inseparable vowel pair

Fig. 1. Summary of hyphenation and syllabication rules.

1. Read the Table of Suffixes and the Table of Exceptions.
2. Read a word.
3. If the word is an exception then read the hyphenation from the exception table and exit; else
   3.1 for all the letters in the word
      1) If a consonant follows a vowel, insert a hyphen after the vowel.
      2) If a vowel follows a vowel, insert a hyphen between them. Save the position of the vowel pair.
      3) If syllabication is required, check if the word starts with *mp, nk, nn* or *nt*. If so, insert a hyphen after the *m* or *n*.
      4) If hyphenation is required and the word ends in *ng*, remove the hyphen preceding the *n*.
      5) If a consonant follows a vowel or a vowel follows a consonant, then do nothing.
   3.2 Resolve double vowels: for each double vowel in the word, hyphenation resolved is set to false.
      1) If the part of the word from the second vowel in the vowel pair onwards is in the suffix table (e.g. *beilwego*, -ilwego), then hyphenation resolved is set to true.
      2) If hyphenation resolved = false and this double vowel starts at position 2 in the word and the word starts with a prefix in the prefix table, then hyphenation resolved is set to true.
      3) If hyphenation resolved = false then check if the double vowel is non-separable. If so, then remove the separating hyphen.
4. If there are any more words to hyphenate, repeat from step 2.

Fig. 2. Algorithm for hyphenation and syllabication of North Sotho.

mented the algorithm caters for punctuation, and if a word already had a hyphen in it before being hyphenated, the program prints this word with an equal sign (=) in the place of the existing hyphen. (See kudukudu below.) Puku ya Merapelo. . .[10] was used as the benchmark for most of the testing; the majority of the words and phrases in it are actually excerpts from the Bible, so testing was performed using 'Biblical quality' text. The table containing hyphenated words is followed by the same words processed by syllabication.

Hyphenated words

'Mma'†	ga-mmo-go	go-tee	se-swaing
maa-tla	be-e-tše-go	i-tše-e-la	se-swai
le-fee-la	nkgwa-na	ba-ti-a-ko-ne	Di-e-ba-nge-di
Kwang	nte-ka	be-i-lwe-go	Tii-še-tšo
e-ba-nge-di,	tse-nywang	ei-tše	Bo-i-po-bo-lo
ngwa-di-lwe-go	hwe-tše	eu-pše	Di-kga-u-ge-lo
Mo-kge-thwa	kwe-na	le-i-neng	me-la-ong
tli-šwe	di-sa-kra-me-nte	ta-e-lo	la-o-la
tu-mi-šeng	pho-lo	I-si-ra-e-le	se-a-tleng
di-phe-to-go	ku-du = ku-du	ye-o	Ba-a-po-sto-la
swa-ne-tše	mpe-tha	Le-na-ne-ong	Mo-a-hlo-di
mpsha-fa-tša	khu-no-llo	Ma-i-ko-ndi-jo	

Syllabicated words

'Mma'†	ga-mmo-go	go-tee	se-swai-ng
maa-tla	be-e-tše-go	i-tše-e-la	se-swai
le-fee-la	n-kgwa-na	ba-ti-a-ko-ne	Di-e-ba-nge-di
Kwa-ng	n-te-ka	be-i-lwe-go	Tii-še-tšo
e-ba-nge-di,	tse-nywa-ng	ei-tše	Bo-i-po-bo-lo
ngwa-di-lwe-go	hwe-tše	eu-pše	Di-kga-u-ge-lo
Mo-kge-thwa	kwe-na	le-i-ne-ng	me-la-o-ng
tli-šwe	di-sa-kra-me-nte	ta-e-lo	la-o-la
tu-mi-še-ng	pho-lo	I-si-ra-e-le	se-a-tle-ng
di-phe-to-go	ku-du = ku-du	ye-o	Ba-a-po-sto-la
swa-ne-tše	m-pe-tha	Le-na-ne-o-ng	Mo-a-hlo-di
m-psha-fa-tša	khu-no-llo	Ma-i-ko-ndi-jo	

Conclusion

The algorithm presented finds every possible hyphenation point of any North Sotho word, given no semantic or syntactic knowledge of the word. As hyphenation and syllabication are similar they were both presented, although hyphenation is more likely to be used in a word processor. The 1951 – 1985 version of the orthography was used to test the algorithm.

Further research in this area should be done to test the algorithm with the two other orthographies, for completeness. Other exceptions need to be found; in the authors' case, this was done by inspection. Extensions to this algorithm could include developing a North Sotho spelling checker and also a translation facility, as these both rely on the morphology of the words.

Received 21 January 1988.

1. Ocker W.A. (1975). A program to hyphenate English words. *IEEE Trans. Professional Communication,* **PC-18** 2, 78 – 84.
2. Mañas J.A. (1987). Word division in Spanish. *Commun. ACM,* **30** 7, 612 – 616.
3. Gee Q.H. (1987). *Automatic hyphenation of Afrikaans.* Research report, Computer Science Department, University of the Witwatersrand, Johannesburg.
4. Hampton N. (1987). *Hyphenation in Zulu.* Honours project, Computer Science Department, University of the Witwatersrand, Johannesburg.
5. Goslin B. (private communication, 1987), Department of African Languages, University of Pretoria.
6. North E.M. (ed.) (1938). *The Book of a Thousand Tongues.* Harper, New York.
7. Ziervogel D. and Mokgokong P.C. (1971). *Klein Noord-Sotho Woordeboek.* Van Schaik, Pretoria.
8. Mpye J. (private communication, 1987). Counselling and Careers Unit, University of the Witwatersrand, Johannesburg.
9. Nalane H.H. (private communication, 1987). Linguistics Department, University of the Witwatersrand, Johannesburg.
10. Church of the Province of South Africa (1982). *Puku ya Merapelo ya Kereke le Difela le Dinoto.* E.L.D. Trust, Johannesburg.

← *471*

ligsondeerder wat tans ontwikkel word, kan deur onopgeleide persone herstel word. Die herstelwerk word dan beperk tot die vervanging van 'n gedrukte stroombaan (koste ongeveer R15) of die vervanging van die meetkabel met die ligtransistors. Die gebruiksduur van die apparaat is feitlik onbeperk en dit behoort etlike jare te hou.

Vervanging van batterye. Met die nuwe model kan die battery van die kant af ingedruk word. 'n Blinde persoon kan die batterye sonder hulp van 'n siende persoon vervang.

Die personeel van die Technikon Pretoria vervaardig die apparaat kosteloos vir die Nasionale Raad vir Blindes. Daar word slegs verwag dat die Raad vir die komponente sal betaal. Die herstel en instandhouding van onklaar apparaat word ook gratis gedoen. Plaaslike maatskappye kan gerus die projek borg sodat die apparaat kosteloos aan blindes beskikbaar gestel kan word.

Hoe werk die apparaat?

Die uitset van die ligsensitiewe transistors word eers met behulp van bewerkingsversterkers (OP-AMP's) versterk. Die uitset van die twee versterkers word deur 'n derde bewerkingsversterker met mekaar vergelyk. Die derde versterker bekragtig 'n ossillator wat slegs in werking tree indien 'n ligsein ontvang word.

Die apparaat is afhanklik van die verskil in lig en nie op die meting van ligsterkte nie. Die huidige apparaat kan direk na die son gerig word sonder enige oudiosein, maar dit kan enige ligdiode se lig waarneem. Tydens praktiese toetse is die diodes se uitstraling verminder sonder dat enige probleme ontstaan het. □