AUTO CORRECTION OF SETSWANA REAL-WORD ERRORS

Gabofetswe Malema, Boago Okgetheng, Moffat Motlhanka and Goaletsa Rammidi

Department of Computer Science, University of Botswana, Gaborone, Botswana

ABSTRACT

Spell checkers are used to detect and where possible correct spelling errors. Errors are classified as non-word errors and real-word errors. Real-word errors require the consideration of the context of the sentence to detect and correct. Setswana language has several commonly used words which are often misspelled by either separating or merging them. The misspelling results in real-word errors. In this paper we propose contextual rules that look at neighbor words to determine whether the correct word is written as two separate words or merged as one word. For some words the rules require that the parts of speech category of neighbor words be determined whereas some depend on specific neighbor words or position in a sentence. Implemented rules show that the rules are very consistent with a 88% success rate. Our tool only looks at neighbor words and therefore does not look at the context of the whole sentence. Hence, for words that require context of the whole sentence to disambiguate correctly our rules fail. This module can be incorporated into a spell checker to detect and correct real world errors for some words. That is, help users to determine the correct orthography of certain words.

KEYWORDS

Spell checker, real-word errors, dictionary

1. INTRODUCTION

Spell Checking is the process of checking if words in a document are in the vocabulary of a language. A spell checker is a software tool used to alert users to possible word misspellings. Most spell checkers also do auto correction and suggest possible words where possible. They are incorporated in many systems such as text editors, word processors, e-mail, and search engines to improve quality of documents, reduce user effort and improve input quality. There are 2 types of errors; non-word errors and real word-errors[1]. A non-word error is when the input word is deemed not to be part of the language vocabulary. A real-word error is when the input word is part of a language vocabulary but its use in the sentence is inappropriate. For example, *"I was using my uncle's fan to move*". In the sentence, the writer intended to say *"van"* and not *"fan"* but *"fan"* is a valid word in English so a typical spell checker would not catch the error. Such errors are more difficult to detect compared to non-word errors as they require analysis of the sentence context.

Error detection and correction methods include statistical, rule based or hybrids[2][3][4]. Statistical methods such as n-grams look at the use frequency of words together. They pick the most frequent sequence of words. This technique can be applied to any language, however, it requires a large and good representative data set for training. However, even with large data n-grams have a problem of data sparseness. Rule based techniques on the other hand rely of the language specifics. That is, they are language dependent and therefore, need language experts to develop the rules. The main advantage with rule-based approach is that they are more accurate

DOI: 10.5121/ijnlc.2019.8405

in most cases. However, they are also limited. Part of speech (POS) based methods make decision based of the part of speech of the word of interest. However, if the 2 words of interest have the same part of speech then the method fails. Most methods work but with a limited coverage.

Many European languages, such as English, French, and Spanish have sophisticated Spell checkers with some capability to detect some real-word errors. Setswana is the national and official language in Botswana. It is also spoken in neighboring countries of South Africa, Zimbabwe, Namibia and Zambia. There are basic spell checkers developed for Setswana language [5][6]. These spell checkers are basic in the sense that they are designed to catch non-word errors only. In this paper we develop rules to detect and correct real-word errors for some words. The rules are based on neighboring words of the word of concern. The rules show a high correction rate for the document tested.

This paper is organized as follows. Section 2 describes Setswana real word errors with common examples and shows some of the rules that could be used to detect misuse of the words. Section 3 describes the implementation of the corrector and Section 4 discusses performance results obtained. Section 5 concludes the paper.

2. SETSWANA REAL-WORD ERRORS

We are not aware of any literature on analysis of Setswana spelling errors patterns. From our experience writing and reading the language we have observed from Setswana documents that certain common words are often interchanged. These errors could be looked at as cases of orthography in that some users may not know if the correct orthography of such words. That is, the writer consciously writes the word in that form thinking it is how it is written. Setswana has commonly used 1 and 2 syllable words which when separated are two valid words and when merged form a valid word also. Examples of such commonly confounded Setswana words include, *ene|e ne (him/her|it was), ee|e e (yes,the), sele|se le (that one, being)*. For instance, in the sentence "monna *o ne* a lwala" (*the man got sick*) is not the same as "monna *one* a lwala" which does not make sense because of the word "*one*". Both "*one"* and "*o ne"* are valid words in Setswana. Setswana writers often use the two interchangeably. This is a source of real word errors for most Setswana writers. Most of such words are pronouns(*maemedi*), demonstratives(*masupi*) and concords(magokedi) as shown below.

Pronouns (Maemedi)
ene(him/her)| e ne (it was)
bone(them)|bo ne (it was)
lone(it)|lo ne (you were/it was)
one(it)|o ne (it was/s(he) was)
gone(it)|go ne (it was)

Demonstratives (Masupi):
yole(that one)|yo le (the one that ..)
bale(those)|ba le(the ones …/those you …)
sele(that one)| se le (the one that ..)
yoo(that one)|yo o (relative concord)
eo(tha one)|e o (concords)

There are several other words such as

gore(so that/because)| go re(to say)
seka(prosecute)| se ka(don't)
kake(cobra)| ka ke (will not) and others that also are common real word errors.

These words like many others words in Setswana language can be used for different functions. We have looked at different uses of each word from available documents and looked for consistent patterns that could deterministically distinguish the appropriateness of one word over its opposite. Below are some of the rules that we have developed by looking at word usage.
2nd Demonstrative and Relative concord

Table 1. 2nd Demonstrative and Relative concords.

2nd demonstrative	Relative Concord
yoo(that one)	yo o
bao(those ones)	ba o
oo(that one)	o o
eo(that one)	e o
leo(that one)	le o
ao(those ones)	a o
seo(that one)	se o
tseo(those ones)	tse o
loo(that one)	lo o
joo(that one)	jo o

The first table shows 2nd demonstratives and relative concords for several nouns classes. The 2nd demonstrative implies that the object is closer to the person being talked to. For example, the 2nd demonstrative for class one nouns is *'yoo'* where as *'yo o'* is the relative concord for the same class.

3rd demonstratives

Table 2. 3rd Demonstrative and indirect relative concords

3rd Demonstratives	indirect relative concords
yole	yo le
bale	ba le
ole	o le
ele	e le
lele	le le
ale	a le
sele	se le
lole	lo le
jole	jo le

yoo can end a sentence.
yoo can be followed by o and a verb.
yo o can be followed by a verb ending with *–ng* or *o* plus *verb+ng*

sele(that one) vs se le
sele can end a sentence but *se le* cannot.
sele cannot be followed by a quantitative but *se le* can.
se se neng is only followed by *se le* and not *sele*
se ne is only followed by *se le* and not *sele.*

ene(him/her) vs e ne(it was)
ene can end a sentence and *e ne* cannot.
words that immediately follow *e ne* are *ya re and e* only.
le should be followed by *ene* and not *e ne.*
ene can be followed by a comma, question mark, semi colon and period but *e ne* cannot.

one(it) vs o ne(he/she/it was)
one can end a sentence but o ne cannot.
o ne should be followed immediately by o or a. one can also be followed by o and a plus others such as wa (possessive concord), demonstrative concords(oo,ono,ole), noun.

ee(yes) vs e e(relative concord)
ee can be followed by comma, semicolon and period.
e e cannot start or end a sentence.
e e can be followed by adjectives.

sena(after) vs se na(not having)
se na is followed by a noun.
sena is always followed by '*go*' then a verb.

seka(being accused)|se ka (do not)
seka is a verb.
se ka is followed by concord plus a verb.

kake(cobra; it's a noun) vs ka ke(cannot)
ka ke cannot be immediately followed by *e.*
ka ke cannot start a sentence.
ka ke is immediately followed by a possessive concord.
Words like *ga e, ga di, ga bo, ga o, o sa, bas a, ke se* are some of the words that cannot be followed by *kake.*

gore(so that,because) vs go re(to say)
go re cannot be followed immediately by *a,o, re, ke , yo, ere, wa, fa , bo, la,*
there shouldn't be *go ne* before gore.
There shouldn't be *e se,* or *jwa, ga se* before go re.

For a complete list of Setswana concords refer to [7].

3. REAL-WORD CORRECTOR IMPLEMENTATION

From the rules it shows that we need a parts of speech tagger to determine whether some words are nouns of verbs. The corrector scans a sentence for one the words above and once found it uses the rules and the parts of speech tagger where necessary to determine whether the word is correct as it or its opposite should be used instead. We use the morphological analyzer developed in [8] as a parts of speech tagger.

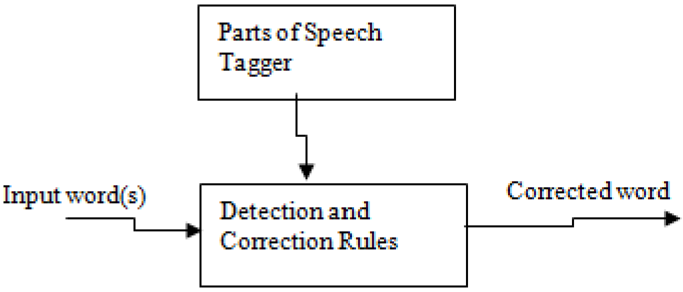

Figure 1. Block Diagram of Proposed Real-Word Corrector

4. PERFORMANCE ANALYSIS

It has to be noted that in this experiment the error detection rate and the correction rate are the same since we have a single predetermined correction if a spelling error is detected. The module was given a 10 000 word document with words of interest. The tables below show the number of words in the document and the number of times it was interpreted correctly. We used the morphological analyze from [8] to determine whether a given word is a noun or a verb.

Table 3. Number of errors detected and corrected

words	number of appearances	Appropriate detection
sele|se le	24	21
ene|e ne	29	28
one|o ne	43	38
ee|e e	40	38
sena|se na	11	10
kake|ka ke	17	14
gore|go re	18	11
seka|se ka	7	5
yoo|yo o	37	34

The table shows that words/rules that do not require their neighbors to be tagged are more successful that those that require tagging. The corrector failure are due to the POS tagger failures and omission of some rules.

5. CONCLUSIONS

We have developed and implemented rules for determining the correct orthography of some words that lead to errors in Setswana. Developed rules look at neighboring words and in some cases use the classification of the part of speech tagger. The results show that in most cases the correct orthography of the words can be determined by use of rules. However, in some cases the accuracy of the rules depend on the performance of the part of speech tagger. These rules can be incorporated into a spell checker. For some words the rules are very accurate which means in those cases the spell checker could do autocorrecting of the words. This study looked at some of the words, more work needs to be done on other words and more testing to exhaust all possible positions (or function) in which these words can appear in a sentence or phrase.

REFERENCES

[1] Dr. G. Malema is a Senior lecturer at the Department of Computer Science, University of Botswana. He obtained his PhD Computer Engineering in 2008 from K. Kukich, "Techniques for automatically correcting words in text", *ACM Computing Surveys*, (24(4), pp 277-439, 1992.

[2] P.H Hema & C. Sunitha, "Spell Checker for non-word Error Detection: Survey", International Journal of Advanced Research in Computer Science and Software Engineering, Vol 5, Issue 3, March 2015.

[3] Graeme Hirst and Alexander Budanitsky, "Correcting real-word spelling errors by restoring lexical cohesion", Natural Language Engineering, 11(1): 87—111 2005

[4] Mashod Rana, Mohammad Sultan and M.F Mridha," Detection and Correction of Real-word Errors in Bangla Language", International Conference on Bangla Speech and Language Processing September 2018.

[5] D J Prinsloo and Gilles-Maurice deSchryver, "Non-word error detection in current South African Spellcheckers". South African Linguistics and Applied Language Studies, 21(4):307—326 2003

[6] Leon Grobbelaar,"A study on creating a custome South Sotho Spelling and Correcting Software Desktop Application", Master of Technology Dissertation 2007, Central University of Technology, Free State, South Africa.

[7] Mogapi, K, "Thuto Puo ya Setswana", Longman Botswana, 184, ISBN:0582 619033

[8] Malema G, Motlogelwa N, Okgetheng B, Mogotlhwane O, "Setswana Verb Analyzer and Generator", International Journal of Computational Linguistics (IJCL), Vol 7, issue 1, 2016

AUTHORS

Dr. G. Malema is a Senior lecturer at the Department of Computer Science, University of Botswana. He obtained his PhD Computer Engineering in 2008 from The University of Adelaide. He has been working on Automation tools for Setswana for the past 5 years.

Mr. Boago Okhetheng is MSc Computer Science student in the Department of Computer Science, University of Botswana. He graduated with a BSc Computer Science from the University of Botswana in 2015.

Mr. Moffat Motlhanka is MSc Computer Science student at the University of Botswana, Department of Computer Science. He graduated with a BSc Computer Science from the University of Botswana in 2018.

Ms Goaletsa Rammidi is a Lecturer in the Department of Computer Science, University of Botswana. She has MSc in Computer Science.