SETSWANA PART OF SPEECH TAGGING

Gabofetswe Malema, Boago Okgetheng and Moffat Motlhanka

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

ABSTRACT

Part of speech tagging is one of the basic steps in natural language processing. Although it has been investigated for many languages around the world, very little has been done for Setswana language. Setswana language is written disjunctively and some words play multiple functions in a sentence. These features make part of speech tagging more challenging. This paper presents a finite state method for identifying one of the compound parts of speech, the relative. Results show an 82% identification rate which is lower than for other languages. The results also show that the model can identify the start of a relative 97% of the time but fail to identify where it stops 13% of the time. The model fails due to the limitations of the morphological analyser and due to more complex sentences not accounted for in the model.

KEYWORDS

Setswana, part of speech tagging, morphological analysis

1. INTRODUCTION

Part of speech tagging is process that identifies parts of speech in a sentence for a given language. It is considered to be one of the fundamental stages of natural language processing for any language. It is a pre-processing stage for advanced applications such as machine learning, translation, and grammar checking [1]. Studies in this area are classified in three main approaches of statistical, rule based and hybrid [1][2][3][4]. Statistical methods are provided with learning/tagged data which trains the tagger. The approach has been shown to give good results in the region of 90's for many languages [2]. However, the approach works well if there is a good training data. The rule based approach applies the language's rules to identify parts of speech. Although the rule based method relies heavily on knowledge of the language it can give better accuracy and feedback [2]. The hybrid approach combines the benefits of statistical and rule based approaches. Initially words are tagged based on statistical approach and if there are words that need disambiguation language rules are used to resolve them [4].

The complexity of tagging varies from language to language. In some languages most tags are just single words or tokens separated by space which makes it easy to identify. However, in some languages the problem is nontrivial. For example in which is Setswana written disjunctively, a few words are grouped together to form a part of speech in some cases. There are few studies which have highlighted the complexity in Setswana part of speech tagging and tokenization. In [5] a statistical approach is used to tag Northern Sotho which is close to Setswana language. The study obtained a performance of 94%. The study however does not indicated whether compounds parts of speech were included or not. A finite state machine approach is used [6] to analyse Setswana verb morphology which also obtained a 94% rate. The tokenization problem is related to this study in that for one to perform tagging they also need to tokenize. In [6] the main aim is to tokenize or tag verbs. In both studies performance results are high and promising. However, the results are not good enough to lead to a completely automated part of speech tagger. Studies do not offer general approaches to single words (tokens) and compound tokens or parts of speech.

DOI: 10.5121/ijnlc.2017.6602

Setswana is a low resourced language and has limited resources in the form of tagged text. We therefore choose to use the rule based approach to perform part of speech tagging. Setswana is a Bantu language spoken in Botswana, South Africa, Zimbabwe and Namibia. There are few automation tools for Setswana language. For Setswana to experience an explosion of NLP applications like other languages, basic tools such as analyzers and part of speech taggers need to be developed. In this study we present a way to identify the relative (*leamanyi*) and possessive *(lerui*) which are one of the common parts of speech.

2. SETSWANA PARTS OF SPEECH

Sentences in a language are made up of parts of speech which include nouns, pronouns, adverbs, adjectives and verbs. Each part plays a role in giving the sentence its meaning. Depending on the language a sentence has a structure based on the language grammar. Setswana uses a disjunctive orthography. That is, in some cases a few tokens (words) are written separately but are only meaningful as a unit. Individual words play different functions in a sentence depending on words around them. This possess a great challenge as a word could have many functions as also explain in [6]. For example the word *ba* could be:

1) a possessive concord: batho *ba* gagwe (his people)
2) a relative concord: *ba ba* lemang (those planting)
3) a pronoun: *ba* a tsamaya (they are going)
4) a demonstrative: batho *ba* (these people)

Table 1. Setswana relative and possessive concords according to noun class [8].

noun class	prefix	Relative concord	Possessive Concord
1	Mo-	yo o	Wa
2	Ba-	ba ba	Ba
3	Mo-	o o	Wa
4	Me-	e e	Ya
5	Le-	le le	La
6	Ma-	a a	A
7	Se-	se se	Sa
8	Di-	tse di	Tsa
9	N-	e e	Ya
10	DiN-	tse di	Tsa
11	Lo-	lo lo	lwa
14	Bo-	jo bo	jwa
15	Go-	mo go	Ga

In the examples above the word *ba* plays a different role depending on the words around it. In Setswana these words are in almost every sentence and they include concords, pronouns and demonstratives. This is particularly the case in adjectives, relatives and adverbs [7][8]. They are made up of a concord which depends on the noun class and the root which can be other parts of speech. In this study we investigate how relatives and possessives can be identified in a sentence. Table 1 below shows relative and possessive concords according to noun class. Based on the sentence examples from Setswana text and literature we have derived general constructs for relatives and possessives.

After investigating how relatives are used in Setswana sentences we derived the following constructs.

cc + X, where cc is one of the possible concords from table 1 and
X can be verb+ng, negative + verb+ng, tense + verb+ng, adjective, locative adverbs

Examples are

yo o lelang(the one crying)
yo o tla lelang(the that will cry)
yo o thata (the tough one)
yo o kwa godimo (the one on the top)
yo o sa leleng(the one not crying)

We have also derived constructs for possessives as follows;

cc + X, where cc is one of the possible concords from table 1 and
X can be a pronoun, noun, relative, possessive, adjective, demonstrative

Examples are

ya bone (theirs)
ya Thabo (Thabo's)
ya wa Thabo (its for Thabo's wife,car etc)
ya ele (for that one)
ya yo moleele (for the tall one)

From these constructs we derived a finite state diagram which is then represented as a two-dimensional array showing all the transitions from the time a concord is detected to the final state when a relative is positively identified. It has to be noted that the constructs above are not exhaustive. There are more complex indirect relatives and possessives. In this study we have looked at a few direct constructs to demonstrate the use of state machines for identification of compound Setswana parts of speech.

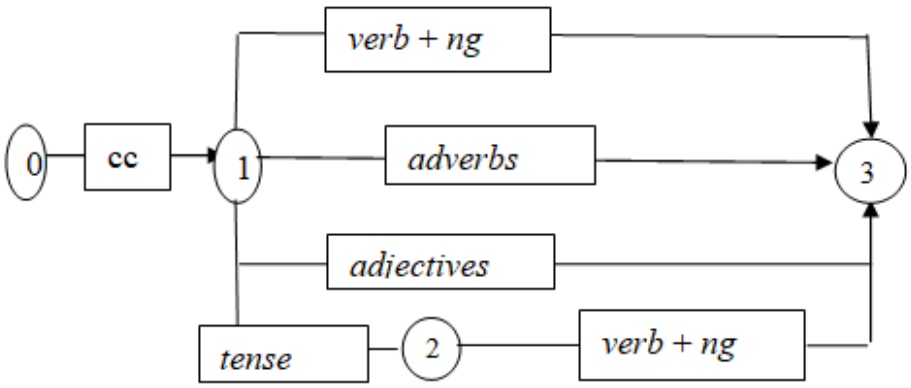

Figure 1. Compressed relative state diagram.

Figure 1 shows a compressed state diagram for identifying a relative part of speech. Starting at state 0 the machine waits for an appropriate relative concord to move to state 1 at which it can accept a few inputs to move to the accepting state 3. The diagram is compressed in that adverbs

and adjectives are also state diagrams. In Setswana descriptive parts of speech can be formed using other parts of speech including other descriptive parts of speech. For example,

e e fa nokeng (the one at/by the river)

This example is a relative which is created using a locative adverb. *e e* is a relative concord for noun class 4 (Table 1) and *fa nokeng (at/by the river)* is a locative adverb. Therefore the whole phrase is classified as a relative.

The equivalent state transition table is as shown in Figure 2.

	cc	verb	tense	adjective	adverb	
0	1					
1		3	2	3	3	
2		3				

Figure 2. State transition table for relative state diagram in figure 1.

The state table is implemented in java as a 2D array. The numbers in the table cells indicate the next state. At each state the transition is determined by the next input. If the transition lands in an empty cell it means the tokens read so far are not forming a relative. We developed the possessive state diagram and transition table the same way as we did above for the relative.

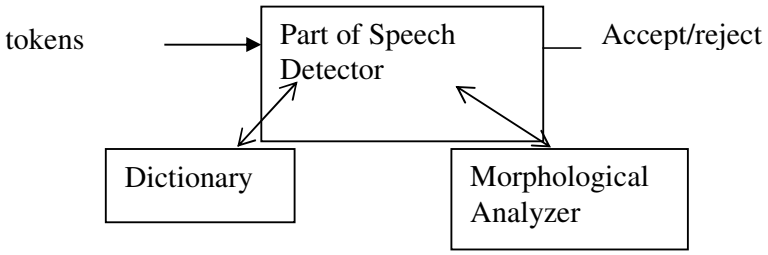

Figure 2. Block diagram of proposed Tagger.

3. PROPOSED PART OF SPEECH TAGGER

Based on the derived finite state transition table derived in the section above, we developed a tool that detects and identifies a relative or a possessive in a sentence as shown as Figure 2. The transition tables are implemented in java as 2D arrays. Concords are stored in another 2D array as shown in Table. The tool uses a dictionary and a morphological analyser. The dictionary has a list of tagged words. That is, identify whether a given token is a pronoun, noun, verb, enumerative, demonstrative and others. After identifying a valid concords the tool needs to classify the next token to determine the next state in the table. The dictionary does not contain all words in their different forms. The morphological analyser is therefore used to reduce words which are feed back into the dictionary for identification and classification. In Setswana verbs can be modified in to many forms including intensive, passive, intensive, reciprocal, neuter, plural, reflexive and others. Nouns also take different forms such as plural, locative and diminutive. Since the dictionary will not have all the words with their derivatives we use the morphological analyser developed in [9] to help in classifying derived words. For example, given a token *berekile (worked),* the dictionary will only have the root form of the word which is *bereka (work).* The morphological analyser will reduce *berekile* to *bereka* which the dictionary will identify as a

verb. The dictionary and the morphological analyser are actually implemented as one module. The tagger scans a sentence from left to right until it detects a possible start of a relative or possessive concord. Once that is detected, the tagger traverses the corresponding sate table until it accepts or rejects the input.

3. PERFORMANCE RESULTS

The proposed tool was tested with a 10 page Setswana document. The results show an overall performance rate of 82%. From the text, 77 relatives were manually identified out of which 65 were identified by the tool resulting with a performance rate of 84% for relatives. The tool could detect 97% of relatives but fails in some cases to detect where they stop.

Also from the test text, 111 possessives were manually identified out of which 89 were correctly identified giving a performance rate of 80%. The tool could detect 98% of possessives but fails in some cases to detect where they end. It has to be noted that this results are only for direct relatives and possessives and do not include indirect ones. From the results analysis it shows that the tool fails to classify some words correctly due to limitations of the dictionary and the morphological analyser. We noticed that in some cases the words could be classified as both adjectives and nouns but our dictionaries in some cases have only one classification. It will be important to develop a tagged dictionary specifically for the purposes of tagging. The tool also fails to correctly identify the end of some relatives and possessives especially the long ones which are made up of other parts of speech. A relative could be made up of other parts of speech such as locative adverbs. In our implementation, other compound parts of speech were not fully defined. The developed rules are limited in identifying complex constructs of these compounds parts of speech. We found out in some cases these parts of speech are made up of three to five parts of speech in recursive manner. That is, a possessive can be created using a relative and that possessive can be created using a relative or some other part of speech.

3. CONCLUSIONS

The disjunctive orthography of Setswana language makes it challenging to automate tokenization and part of speech tagging. We developed finite state diagrams for relatives and possessives and we implemented them in Java. The tagger is supported by a dictionary and a morphological analyser. From the test data the tool detects shorter parts of speech well compared to longer ones. Overall the tool achieved 82% identification rate. The tool can be further improved by improving the dictionary, the morphological analyser and adding other parts of speech which are used with the parts of speech. The tagger heavily relies on the dictionary and morphological analyser to identify tokens. The approach could also be applied to other compounded parts of speech in a similar way.

REFERENCES

[1] Charniak E (1997) “Statistical techniques for Natural Language parsing”, AI Magazine, 18(4), pp.33-44

[2] Brants T (2000) “A statistical part of speech tagger”, PANCL’00 Proceedings of the sixth conference on applied natural language processing Association for Computational Linguistics

[3] Brill E (1992) “A simple rule based part of speech tagger”, In Proceedings of the third conference on Applied Natural Language processing, ACL, Trento, Italy

[4] Brill E (1995) “Transformation Based Error-Driven Learning and Natural language Processing: A case study in Part of Speech Tagging”, Computational Linguistics

[5] Faa G, Heid U, Taljard E & Prinsloo D (2009) “Part-of_Speech tagging of Northern Sotho: Disambiguating polysemous function words”,Proceedings of the EACL, 2009 Workshop on Language Technologies for African Languages – Aflat 2009, pages 38—45, Athens Greece, 31 March 2009

[6] Pretorius L, Viljoen B, Pretorius R and Berg A.(2009) "A finite state approach to Setswana verb morphology", International Workshop on finite state methods and natural Language Processing FSMNLP 2009: Finite State Methods and Natural language Processing, pp. 131 – 138

[7] Cole D T, "An Introduction to Tswana grammar", Longmans and Green, Cape Town.

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

[9] 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 3 years.

Mr. Boago Okhetheng is an assistant research. He graduated with a BSc Computer Science from the University of Botswana in 2015.

Mr. Moffat Motlhanka is currently a forth year Computer Science student at the University of Botswana.