A Grapheme-Based Approach for Accent Restoration in Gĩkũyũ

Peter W. Wagacha*, Guy De Pauw† and Pauline W. Githinji*

*School of Computing & Informatics
University of Nairobi
Box 30197-00100, Nairobi, Kenya
waiganjo@uonbi.ac.ke, pnishus@yahoo.com

†CNTS - Language Technology Group
University of Antwerp
Universiteitsplein 1, 2610 Antwerpen, Belgium
guy.depauw@ua.ac.be

Abstract

The orthography of Gĩkũyũ includes a number of accented characters to represent the entire vowel system. These characters are however not readily available on standard computer keyboards and are usually represented as the nearest available character. This can render reading and understanding written texts more difficult. This paper describes a system that is able to automatically place these accents in Gĩkũyũ text on the basis of local graphemic context. This approach avoids the need for an extensive digital lexicon, typically not available for resource-scarce languages. Using an extended trigram based approach, the experiments show that this method can achieve a very high accuracy even with a limited amount of digitally available textual data. The experiments on Gĩkũyũ are contrasted with experiments on French, German and Dutch.

1. Introduction

The Gĩkũyũ language is spoken by over five million people and is the most commonly spoken Bantu language in Kenya, second to Swahili. Like most Bantu languages, the orthography of Gĩkũyũ is straightforward with mostly a one-to-one phoneme to grapheme mapping. To represent all seven vowels, it therefore needs to introduce two extra diacritically marked graphemes into its alphabet: the cardinal vowels *ĩ* and *ũ* which represent distinct phonemes from those represented by the unmarked *i* and *u* graphemes. These characters are however not readily available on standard computer keyboards and are therefore often represented as the nearest available character. This common practice of using the unmarked graphemes to represent both phonemes can render reading and understanding written texts more difficult, since it involves a disambiguation process on the part of the reader. This is especially problematic in Gĩkũyũ and other closely related languages like Kĩembu, Kĩmerũ and Kĩkamba, where in a typical text more than 50% of the words can be marked with one or more diacritic. In this paper we investigate how these accents can be automatically restored using machine learning methods.

For other languages using diacritics, such as German or French, this task can typically be handled by a simple lexicon lookup procedure that translates words without accents into the properly annotated format. This type of information source is however not digitally available for most African languages, many of which make extensive use of accented characters. We propose a machine learning approach that tries to predict the placement of accents on the basis of local graphemic context and contrast it with a traditional dictionary lookup approach for Dutch, German and French. We improve on existing approaches by applying trigram based classification to make the output of the system more robust.

The paper is organized as follows: we first look at previous work on accent restoration. Next we discuss the languages and datasets used in this paper and describe some baseline models for accent restoration. We then outline the experiments with the grapheme-based machine learning method and conclude with some pointers to future work.

2. Previous Work

Most of the work on accent restoration tackles both the actual task of retrieving diacritics of unmarked text, as well as the related tasks of part-of-speech tagging and word-sense disambiguation. Yarowsky (1994) compares a number of corpus-based techniques to this end for Spanish and French. Although complete accent restoration would ideally involve a large amount of syntactic and semantic disambiguation, this type of linguistic analysis can typically not be done for resource-scarce languages. Moreover, the accent restoration methods presented in Yarowsky (1994) and related research efforts (Tufiş and Chiţu, 1999; Simard, 1998) rely heavily on lexicon lookup procedures and are therefore not applicable to our target language, Gĩkũyũ.

Mihalcea (2002) presents a diacritics restoration system that uses machine learning methods and operates on the level of the grapheme. It is specifically geared towards languages for which no large electronic dictionaries are available. The system is applied to Romanian (like Tufiş and Chiţu (1999)) and achieves accuracy scores of up to 99% on the grapheme level. While it establishes an interesting method that is in theory applicable to all languages that use diacritics, Mihalcea (2002) does not compare the system on other languages, nor are accuracy scores on full words reported. In this paper we compare a similar grapheme-based method for Gĩkũyũ, Dutch, German and French and evaluate it on the word level, since this is the level of description we feel this type of processing task for text corpus development needs to be considered.

3. The Datasets

The focus in this paper is on Gĩkũyũ, since it provides an interesting challenge for accent restoration: there is no digitally available dictionary for this language, nor extensive text corpora. We therefore developed a fully diacritically marked text corpus of about 14,000 words. The corpus comprises of short stories and letters, poems, proverbs and riddles, songs, bible verses and other religious material which have been scanned, OCR-ed and manually corrected. From this corpus we extracted a lexicon of about 4,500 unique word tokens. Most of the available material is in the religious domain and does therefore not reflect everyday language use. For this particular task however, operating on the level of the grapheme, this does not constitute a big problem.

We contrast the experiments on Gĩkũyũ with accented languages for which extensive lexicons are available: Dutch, French and German. This provides some insight into how well the grapheme-based method can approach the accuracy of the optimal lexicon lookup procedure and establishes an upper bound in our evaluation. Similarly to the Gĩkũyũ dataset, we extracted a 45,000 word lexicon from a 23 million word corpus of French newspaper text (1995 volume of Le Monde). For German and Dutch, we used the readily available orthographic databases from CELEX (Baayen et al., 1993) each providing a lexicon of about 330,000 unique word forms.

For the 10-fold cross validation experiments described in this paper, the lexicons were randomly divided into ten partitions, providing a held-out test set for each set of experiments. Eight partitions were used to train the respective models, while the other two partitions were used for tuning and testing respectively. This allows us to measure the performance of the systems on previously unseen words and presents a worst-case scenario in which lexicon lookup is inherently impossible. In addition, we present results of experiments on plain text documents, to evaluate the performance of the system as a practical application for text corpus development.

Table 1 provides an indication of the relative difficulty of the diacritic placement task for each of the languages, by presenting the relative number of accented words in both the lexicon and a typical text. The percentage of accented words in French is in line with those reported in Simard (1998). Dutch counts very few accented words, both in the lexicon and in plain text. While this means that getting a high word accuracy rate for Dutch is trivial, not a lot of positive examples can be found during training. On the other end of the spectrum is Gĩkũyũ with 66.5% percent of accented words in the lexicon, posing a serious challenge for word accuracy rate.

4. Baseline Models

We define two baseline models. The first baseline model identifies candidate graphemes for diacritic marking and chooses the most frequent solution observed in the training set. For French and Dutch for instance these invariably equal to the unmarked characters. This trivial baseline already achieves a very high accuracy for Dutch (Table 3) because of the limited use of diacritics in this language.

	lexicon	**text**
Gĩkũyũ	66.5%	54.6%
French	27.6%	19.7%
German	22.3%	6.8%
Dutch	1.2%	0.3%

Table 1: Percentage of accented words in the lexicon and in running text

	i/ĩ	**u/ũ**
Baseline 1 (Most Frequent)	54.0%	74.7%
Baseline 2 (Lexicon)	74.5%	70.3%
MBL - Grapheme (unigram)	77.6%	86.8%
MBL - Grapheme (trigram)	91.7%	94.0%

Table 2: Grapheme accuracy scores: disambiguation of *i/ĩ* and *u/ũ* graphemes (Gĩkũyũ plain text)

Word accuracy scores for French and German are reasonable, but only 57% of all words in a plain Gĩkũyũ text are marked correctly. Table 2 shows detailed results for the diacritic placement on the two ambiguous graphemes in: *ũ* is the most common graphemic variant, so that the baseline scores reasonably well. Furthermore, the baseline scores show there is an almost equal number of *ĩ* and *i* graphemes. A second baseline model implements the aforementioned lexicon lookup method. In this approach, the training set lexicon is used to translate the unmarked words in the test set into the associated accented words. Trivially, this baseline model fails to score any points on the 10 fold CV experiments (left hand side of Table 3): the test set contains nothing but unknown words unavailable in the training lexicon. Therefore only the results on the plain text test set are relevant for this baseline model.

Particularly for languages with a large training lexicon, this is indeed the baseline to beat. The results show that for Dutch and German, the lexicon lookup model scores quite well. For the former, this is almost a solved problem. Not surprisingly, the much smaller lexicon for French yields a more modest score for the plain text test set. The score for Gĩkũyũ is even lower. For the ”*u/ũ*” disambiguation task, this baseline even underperforms when compared to the trivial baseline 1 method.

5. Grapheme-Based Machine-Learning Approaches

The Gĩkũyũ corpus does not contain enough words to yield a large enough lexicon for a viable dictionary lookup approach. In this paper, we investigate an alternative that redefines the problem as a disambiguation task to be performed on the level of the grapheme, rather than on the word level. This approach has already been attempted for Romanian with a high degree of accuracy (Mihalcea, 2002), but to our knowledge this is the first research effort in this vein for a Bantu language. We hypothesize that even the relatively small Gĩkũyũ corpus is large enough to capture all relevant graphemic contexts for this type of disambiguation task.

L	L	L	F	R	R	R	C
-	-	-	**m**	b	u	r	-
-	-	m	**b**	u	r	i	-
-	m	b	**u**	r	i	-	ũ
m	b	ũ	**r**	i	-	-	-
b	ũ	r	**i**	-	-	-	i

L	L	L	F	R	R	R	C
- - -	- - -	- - -	**- -m**	-mb	mbu	bur	- -m
- - -	- - -	- -m	**-mb**	mbu	bur	uri	-mb
- - -	- -m	-m b	**mbu**	bur	uri	ri-	mbũ
- -m	-mb	mbũ	**bur**	uri	ri-	i- -	bũr
-mb	mbũ	bũr	**uri**	ri-	i- -	- - -	ũri
mbũ	bũr	ũri	**ri-**	i- -	- - -	- - -	ri-
bũr	ũri	ri-	**i–**	- - -	- - -	- - -	i- -

Figure 1: Instances for Grapheme-Based Diacritic Placement Classification of the word "mbũri" (goat). Unigram features (left) and Trigram features (right)

All Words	**Unknown Words**				**Plain Text**			
	Dutch	**French**	**German**	**Gĩkũyũ**	**Dutch**	**French**	**German**	**Gĩkũyũ**
	Dut	Fre	Ger	Gik	Dut	Fre	Ger	Gik
Baseline 1 (Most Frequent)	98.9	71.5	43.9	46.6	99.7	79.7	71.1	57.1
Baseline 2 (Lexicon)	0	0	0	0	99.9	86.5	96.2	74.9
MBL - Grapheme (unigram)	99.5	82.2	91.6	68.0	99.8	88.3	95.3	77.5
MBL - Grapheme (trigram)	99.7	82.8	89.5	72.2	99.5	89.0	94.3	91.4

Table 3: Word accuracy scores

5.1. Unigram Approach

In our approach, the graphemic contexts observed in the training set are used as the information source for the memory-based classifier TiMBL (Daelemans and van den Bosch, 2005). The table on the left in Figure 1 illustrates how the training instances are generated. Using a windowing approach, we record for each grapheme of each word in the training lexicon its left context (disambiguated) and its right context (still ambiguous). For each grapheme we identify a class which in the unigram approach is only made explicit for diacritic candidates *i/ĩ* and *u/ũ*.

The algorithmic parameters of the classifier were optimized on a tuning set and its accuracy was measured on a held-out test set. Table 2 shows that for the *i/ĩ* disambiguation task the classifier improves significantly over baseline 1, while the improvement over the lexicon lookup method is minimal. There is a better improvement for the disambiguation of *u/ũ*. Looking at the overall score for plain text however, we notice that almost one quarter of the words are still not annotated correctly. Accuracy on unknown words seems particularly problematic using the unigram approach and is much poorer than we would generally expect for Gĩkũyũ (Table 3).

Surprisingly, for plain text the unigram grapheme approach overtakes the lexicon lookup method for French. There is only a small decrease for Dutch and German compared to the lexicon lookup approach. These results are encouraging, since they give an indication of the relative accuracy of the grapheme-based approach. Despite these encouraging results, we are still far away from the 99% scores reported by Mihalcea (2002) for Romanian. If we are to use the accent restoration system as a postprocessing tool to minimize manual intervention after corpus compilation, we particularly need to improve on the word accuracy scores.

Predicted Class 1	-	-	m						
Predicted Class 2		-	m	b					
Predicted Class 3			m	b	u				
Predicted Class 4				b	ũ	r			
Predicted Class 5					ũ	r	i		
Predicted Class 6						r	i	-	
Predicted Class 7							ĩ	-	-
Majority Vote:	-	-	m	b	ũ	r	i	-	-

Table 4: Trigram-Based Diacritic Restoration for the word "mbũri" (goat)

5.2. Trigram Approach

We implemented a trigram based approach, in which we record grapheme trigrams using a similar windowing method as in the unigram experiment. The atomic units in this approach are not the single graphemes, but a triplet of graphemes, as illustrated on the right-hand side of Figure 1. This approach has the advantage of capturing larger graphemic contexts, while at the same time still limiting the level of description enough to capture most relevant contexts from a small corpus.

During classification the trigram method effectively provides for each grapheme three separate classification decisions. This is illustrated in Table 4: in this example the classifier has predicted 7 trigram classes for the word "mbũri". These trigram classes are pulled apart and stacked. Using majority voting after classification, we select the grapheme that is predicted at least two times by the trigram classifier[1]. This type of processing has previously been shown to be beneficial for a large number of sequence-based classifi-

[1] A majority is always guaranteed for this type of binary classification.

cation tasks (Daelemans and van den Bosch, 2005).

The results show that the trigram approach provides a substantial increase for the Gĩkũyũ accent restoration task. It significantly outperforms the unigram approach with scores well into the 90% range for the individual graphemes (Table 2). It achieves an impressive word accuracy increase of almost 14% on the plain text test set. The performance increase on unknown words is not as dramatic, but is still substantial. As a tool for Gĩkũyũ corpus construction, the trigram approach seems a viable solution now, with only 1 out of 10 words that need correcting. Many of the remaining problems are caused by ambiguity on the word level and cannot be solved on the level of the grapheme.

There is however a trade-off for Dutch and German, as we observe the unigram system performing better than the trigram approach. Data analysis shows that the trigram approach indeed has a stronger tendency to place diacritics, consequently making more mistakes on words that don't need them. In fact, when evaluating the systems only on words that are supposed to be accented, the trigram approach can be observed to outperform the unigram approach, further illustrating the eager accent placement of the former.

The results also show that the grapheme-based machine learning approach improves on the lexicon lookup approach for Gĩkũyũ. It even approximates the lexicon lookup scores achieved on German and Dutch. It is interesting to observe however, that there is *no free lunch* when it comes to grapheme-based diacritic restoration and that there is a noticeable difference even among related languages.

6. Conclusion and Future Work

In this paper we have presented experiments with a grapheme-based machine learning approach for accent restoration in Gĩkũyũ, German, Dutch and French. The results show that this method can provide accurate diacritic placement, rivaling a standard lexicon lookup approach, especially when confronted with languages for which relatively few textual resources are available. We are confident that the accent restoration system proposed in this abstract will significantly speed up corpus development for Gĩkũyũ as it provides an effective post-processing tool for OCR, as well as a valuable aid for human annotators during transcription. Furthermore, the diacritic placement method for Gĩkũyũ can also be used to process related Bantu languages like Kĩembu, Kĩmerũ and Kĩkamba and can therefore help us to digitally preserve these resource-scarce and/or endangered Bantu languages in their full orthographic form.

The experiments presented in this paper were conducted with the memory-based learning classifier TiMBL. This nearest neighbor approach seems very well suited to this type of linguistic processing task. For future work however, we will also take a look at other machine learning methods, including Support Vector Machines, Maximum Entropy Learning, Hidden Markov Modeling, Artificial Immune Systems and Naive Bayes approaches. We already conducted some preliminary experiments with Support Vector Machines, which have been found to perform well on a variety of machine learning tasks (Wagacha et al., 2004).

As previously mentioned, accent restoration is a task that cannot be fully solved on the level of the grapheme. With its extensive use of diacritics, ambiguous word forms are plentiful in Gĩkũyũ. The ambiguous word form *iria* for example, could be marked as *iria* (English: *milk, ocean, lake* or *those*) or as *ĩrĩa* (English: *that*). The diacritic placement system presented in this paper cannot deal with this type of ambiguity. This particular example could be solved with a bigram language model or part-of-speech tagging. But there are many other examples that would require an accurate word-sense disambiguation method to trigger the correct accent restoration, for example in the case of *irima* (English: *hole*) vs *irĩma* (English: *big hill*). As corpus construction for Gĩkũyũ progresses (De Pauw et al., 2006), we hope to tackle these issues in the future.

Acknowledgments and Demo

The research presented in this paper was made possible through the support of the Flemish Inter-university Council (VLIR), under the VLIR-IUC-UON program.

A demonstration system for Gĩkũyũ diacritic placement can be found at http://www.cnts.ua.ac.be/gikuyu.

7. References

R. H. Baayen, R. Piepenbrock, and H. van Rijn. 1993. *The CELEX lexical data base on CD-ROM*. Linguistic Data Consortium, Philadelphia, PA.

W. Daelemans and A. van den Bosch. 2005. *Memory-based Language Processing. Studies in Natural Language Processing*. Cambridge University Press, Cambridge, United Kingdom.

G. De Pauw, P.W. Wagacha, and K. Getao. 2006. Developing a corpus for Gĩkũyũ using machine learning techniques. In *Proceedings of LREC Workshop: Networking the development of language resources for African languages*, Genoa, Italy.

R. Mihalcea. 2002. Diacritics restoration: Learning from letters versus learning from words. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 339–348, Mexico City, Mexico.

M. Simard. 1998. Automatic insertion of accents in French text. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing*, Granada, Spain.

D. Tufiş and A. Chiţu. 1999. Automatic diacritics insertion in Romanian texts. In *Proceedings of the International Conference on Computational Lexicography (COMPLEX 1999)*, Pecs, Hungary.

P.W. Wagacha, B. Manderick, and K. Getao. 2004. Benchmarking support vector machines using statlog methodology. In A. Nowe, T. Lenaerts, and K. Steenhaut, editors, *Proceedings of Benelearn 2004, Machine learning conference of Belgium and Netherlands*, pages 185–190, Brussels, Belgium.

D. Yarowsky. 1994. A comparison of corpus-based techniques for restoring accents in Spanish and French text. In *Proceedings, 2nd Annual Workshop on Very Large Corpora*, pages 19–32, Kyoto, Japan.