Evaluating a Machine Learning Approach to Sinhala Morphological Analysis

Viraj Welgama
Language Technology Research Laboratory,
University of Colombo School of Computing,
Colombo 00700
wvw@ucsc.lk

Ruvan Weerasinghe
Language Technology Research Laboratory,
University of Colombo School of Computing,
Colombo 00700
arw@ucsc.lk

Mahesan Niranjan
School of Electronics and Computer Science,
University of Southampton,
Highfield, Southampton,
SO17 1BJ, UK
mn@ec.soton.ac.uk

Abstract

In this work, we report an evaluation of a popular morph segmentation algorithm using machine learning on the task of morphologically analyzing Sinhala. We summarize the development of a Sinhala Gold Standard dataset for this purpose and report evaluation results on it. The results indicate that 35% of the words are decomposed in a linguistically accurate manner, while for 50% of the words the correct stems are identified. We also discuss some language specific issues arising from the experience of applying a machine learning approach to morphological analysis of Sinhala. This work can be treated as a benchmark on adopting machine learning approaches for Indic language morphology since no previous attempts can be found in the literature.

1 Introduction

Identifying morphemes, the smallest meaningful units of a word is very essential for modern Natural Language Processing tasks. Tasks such as Speech Recognition, Machine Translation, Information Retrieval and Statistical Language Modeling among others are benefited by good morphological analyzers especially for morphologically rich languages. There are two major approaches for identifying morph segments of a word namely; *knowledge-based* approaches and *data-driven* approaches. Though very successful, the knowledge-based approaches are very expensive with respect to the human resource they require. As a result, research on morphological segmentation is now moving towards more data-driven approaches, which require less expertise and heuristics, but rely on data.

Automatic morphological analysis of language has been of interest to researchers for several decades. Some of the early work in automatic morph decomposition is its successful use in speech synthesis in the MITalk system (Allen et al., 1987). Many early morphology discovery algorithms focused on identifying prefixes, suffixes and stems rather than attempting to produce morphological analysis (Eg: (Dejean, 1998)). Then (Goldsmith, 2001) used minimum description length (MDL) analysis to model unsupervised learning of the morphological segmentation of European languages using corpora. In this work he tries to produce output that would match as closely as possible with the analysis given by human experts.

Data-driven approaches for word segmentation have achieved considerable success over the last decade with help of incentives such as the Morpho Challenge Competition (Kurimo et al., 2011). This was launched in 2005 to challenge the machine learning community, linguists and specialists in NLP applications to study this field and come together to compare their algorithms against each other (Kurimo et al., 2010). Many agglutinative languages[1] such as Turkish and Finnish highly benefited through this competition since their morphology was more complex than the usual European languages for which morphological parsers were available. Many of the competitors attempted to develop language independent morpheme induction algorithms as they can then be used for many other languages.

Our goal in this paper is to present the results

[1] Words in agglutinative languages are formed by concatenating morphemes

and findings on applying such machine learning approach for morpheme segmentation to the Sinhala language. Sinhala is an Indo-Aryan language spoken by more than 16 million people in Sri Lanka. Sinhala is a highly inflectional language as are many other Indic languages, and like many of them, can be considered as a low-resourced language with respect to the linguistic resources available for NLP.

Among many such machine learning approaches for morpheme segmentation, we selected the algorithm called *Morfessor*, a state-of-the-art algorithm for automatic morpheme induction. Freely available source, easy configuration and ability to train only with a list of raw text are some reasons for this selection.

2 Morfessor

Morfessor is a morpheme segmentation algorithm developed by the organizers of the Morpho Challenge competition. It is excluded from the official competition to avoid bias.

Morfessor takes as input a list of un-annotated words with their corresponding frequencies and produces a segmentation of the word forms observed in the text. The segmentation obtained often resembles a linguistic morpheme segmentation (Creutz and Lagus, 2005b). It has evolved through many successive works carried out by the authors from 2002 to 2006 and has passed through different stages of development as described in ((Creutz and Lagus, 2002), (Creutz and Lagus, 2004), (Creutz and Lagus, 2005a) & (Creutz and Lagus, 2007)).

The authors confidently claimed that Morfessor is language independent. The initial version has been developed using two approaches for unsupervised learning. The cost function of the first model has been derived from the Minimum Description Length principle from classical information theory while the cost function of the second method has been defined as the maximum likelihood of the data given the model (Creutz and Lagus, 2002). The first version was called *Morfessor-Baseline* since it was the base for the later models.

The authors further developed the algorithm for *context-dependent* models called *categories-ML* and *categories-MAP*. The final version of Morfessor is developed by utilizing a probabilistic maximum a posteriori model (MAP). This model builds hierarchical representations for a set of morphs and the aim of using it, is to find the optimal balance between the *accuracy* of the representation and the model *complexity*, which generally improves its generalization capacity by inhibiting overlearning (Creutz and Lagus, 2005b). For the experiments we propose in this work, we used this final version of the algorithm which is freely available for experimentation from the Laboratory of Computer and Information Science of the Department of Computer Science and Engineering at the Helsinki University of Technology, Finland.

3 Defining a Gold Standard for Sinhala

One direct way of evaluating automatic morpheme segmentation algorithms is to compare their output with a pre-defined morpheme definition (referred to as a *Gold Standard*) of a given word. Organizers of the Morpho Challenge competition have used this method as one way of evaluating the algorithms and they have provided some sample Gold Standard definitions for English, German, Turkish and Finnish. Prior to carrying out this experiment, we decided to develop such a resource for the Sinhala language as it would facilitate further experimentation.

Developing Gold Standard morphology definitions for a particular language is a highly challenging task, which needs lots of linguistic expertise and heuristic knowledge. We used a prior effort by (Weerasinghe et al., 2009) as the base to develop the Sinhala Gold Standard Definitions (SGSD). They have identified major POS categories of the Sinhala language and all the sub-structures of each category with a comprehensive list of words for each category. We defined and verified all the derivations for each category, initially for a single word and then used a computer program to generate all the derivations of all other words given in the lexicon described. Figure 1 shows a sample Gold Standard definition for the Sinhala word "එළුවා" (*the goat*), which is from the category *Nouns-Masculine.BackVowel*.

The left-hand side (separated by the colon :) of each morpheme shown, is the morph realization at the particular word, while right-hand side is the definition of the morpheme. ~ stands for the empty morph which is highly utilized in Sinhala Nouns. We defined the Gold Standard definitions in such a way that they can produce the relevant word by simply concatenating the left-hand side morphs.

Word	Definition
එළු *{Goat – Root}*	එළු:එළු_N+RT
එළුවා *{the Goat}*	එළු:එළු_N+RT වා:+SG ~:+DF ~:+NOM එළු:එළු_N+RT වා:+SG ~:+DF ~:+ACC
එළුවාත් *{and Goat}*	එළු:එළු_N+RT වා:+SG ~:+DF ~:+NOM ත්:+CJ එළු:එළු_N+RT වා:+SG ~:+DF ~:+ACC ත්:+CJ
එළුවායි *{is Goat}*	එළු:එළු_N+RT වා:+SG ~:+DF ~:+NOM යි:+FN එළු:එළු_N+RT වා:+SG ~:+DF ~:+ACC යි:+FN
එළුවෙක් *{a Goat}*	එළු:එළු_N+RT වෙ:+SG ක්:+ID ~:+NOM

Figure 1: Sample Gold Standard Definitions for Sinhala Masculine Nouns

On seen in Figure 1, there can be more than one Gold Standard definitions for a single word. There are nearly 50 such derivational forms for Sinahla nouns while the number of different forms for Sinhala verbs reaches 200. However, other main POS categories such as adjectives, adverbs and function words[2] have only few (often just 3) derivational forms.

We created the SGSD version 1.0, by defining the morphs for 435,076 unique Sinhala words obtained from 4 major categories (of the 21 different sub categories) of nouns, 3 major categories (of the 5 different sub categories) of verbs, and all adjectives, adverbs and function words described in (Weerasinghe et al., 2009). Table 1 shows the POS distribution of SGSD version 1.0.

In this research we used a simplified version of the SGSD which only contains morph segmentation boundaries to evaluate the output of the Morfessor algorithm. The evaluation carried out with this simplified version is described in section 5.

[2]Function words consist of Conjunctions, Determinants, Interjections, Particles and Post Positions of Sinhala Language.

POS Category	No of Lemmas	No of Word Types	%
Nouns	7,765	329,152	66.00
Verbs	792	160,138	32.11
Adjectives	2,576	7,503	1.50
Adverbs	245	671	0.13
Function Words	609	1,256	0.25
Total	**11,987**	**498,720**	**100**

Table 1: POS distribution of SGSD version 1.0

4 Preparing the Data Set

We used the Morfessor Categories-MAP, version 0.9.2, (Creutz and Lagus, 2012) as the script to be evaluated. The training data was prepared as described below.

4.1 Preparing the Training Data

We used two sets of data to train the Morfessor algorithm. One set was the full distinct word list extracted from the UCSC[3] 10M words Sinhala Corpus (Weerasinghe et al., 2007) (hereafter referred to as the "Full List") while the other set was the distinct word list having more than a single occurrence in the same corpus (hereafter referred to as the "Restricted List"). The frequency value for each word is also extracted from the same corpus.

Before extracting these lists, we tagged the UCSC 10M Words Sinhala corpus using the predefined list of words described by (Weerasinghe et al., 2009). This was done to get a rough idea about the Part Of Speech (POS) distribution of the training sets. Even though the tagging covered 81.6% words in the corpus, the percentage of unclassified words is high in terms of the distinct words in the corpus. Table 2 shows the basic statistics and POS distribution of both lists, used in the training.

	Full List	Restricted List
No. of Word Types	440,020	210,950
Nouns (%)	17.0	25.2
Verbs (%)	4.6	6.9
Unclassified (%)	74.3	61.6

Table 2: POS distribution of SGSD version 1.0

Some initial experiments with the Morfessor algorithm showed that it performs better for the Romanized version of the Sinhala words rather than

[3]University of Colombo School of Computing

for the Sinhala script itself. Unlike Roman scripts, the Indian scripts including Sinhala use vowel modifiers to denote the vowel associated with a consonant. However, in contract, most of such Indian scripts do not use any modifier to denote vowel 'a', but use a modifier to drop the vowel 'a' instead. This confusion may cause Morfessor to under-perform with the Sinhala script. Therefore both lists obtained from the corpus were romanized using a pre-defined scheme before being used as the training data for the algorithm. The scheme used to romanize Sinhala scripts is shown in the Appendix.

4.2 Defining the Test set

From the two lists of words extracted from the UCSC 10M Words Sinhala corpus for the test data, we selected a subset of only those that have a Gold Standard definition in the defined SGSD (described in section 3) for testing. 69,735 words were so extracted from the corpus together with their frequencies under the above condition. Table 3 shows the POS distribution of the selected list for testing the algorithm.

POS Category	No of Word Types	%
Nouns	48,650	69.76
Verbs	16,966	24.33
Adjectives	2,908	4.17
Adverbs	314	0.45
Function Words	897	1.29
Total	**69,735**	**100**

Table 3: POS distribution of the Test Data

We assumed that the POS distribution of the test data fairly represents the real POS distribution of words in the Sinhala language. We then obtained the distribution of the number of morphemes over each POS category. This helped to identify the actual behavior of each category in our error analysis. Table 4 shows the percentage values of the distribution of morphemes.

As we did for the training set, we romanized the test list before using it to test the performance of the algorithm.

5 Evaluation

The authors of the Morfessor algorithm have used the F-measure as the basic evaluation criterion of their system. Since unsupervised algorithms are unable to predict morpheme labels as they appear in pre-defined Gold Standards, their evaluation has to be based on what words forms share the same physical morphemes. They produced large number of sample word pairs from both the proposed analysis and the Gold Standard, such that both words in the produced pair have at least one morpheme in common (Kurimo et al., 2010). They have defined both Precision and Recall based on these pairs and then calculated the F-measure to compare the results. (Creutz and Lagus, 2006) have compared the performance of the Morfessor algorithm for English, Finnish and Turkish based on this F-measure.

POS Category	#[a]=1 (%)	#=2 (%)	#=3 (%)	#=4 (%)	#=5 (%)
Nouns	13.6	26.8	45.1	13.8	0.7
Verbs	05.6	68.4	16.6	9.1	0.3
Adjectives	60.8	39.2	-	-	-
Adverbs	73.9	26.1	-	-	-
Function Words	65.0	35.0	-	-	-

Table 4: Morphemes distribution of each POS category

[a]Number of Morphemes

As the organizers of the Morpho Challenge competition, the authors have used this F-measure based evaluation criterion to evaluate the performance of other morph segmentation algorithms that participated in the competition. (Kurimo et al., 2010) have listed a summary of these results for English, Finnish, German and Turkish for the best algorithms presented in the 2010 competition.

We however, decided to evaluate this algorithm by directly comparing the output with the defined SGSD due to the following reason. The primary objective of our work is to evaluate how a state-ofthe-art machine learning algorithm performs on Sinhala morphology, and not to compare our accuracy with that of other languages. Therefore the direct comparison with the pre-defined Gold Standard definitions is more appropriate and it helped us to get the exact accuracy than by using the F-measure.

5.1 Metrics of Evaluation

We defined three aspects to compare the Morfessor output with the SGSD. One aspect is to check how many words are exactly segmented by the al-

gorithm, as it is in the SGSD (referred to as the "Exact" in the column title of the result tables). This can be claimed as the ultimate accuracy of the algorithm since it gives the number of words which produced the linguistically correct segmentation.

The other aspect of measuring accuracy is to compare only the stems of the segmented words with the SGSD (referred to as the "Stem" in the column title of the result tables). This is a reasonable measure to estimate how well the Morfessor algorithm can perform as a stemmer for the Sinhala language. The third and final aspect is to compare the number morphemes predicted by the algorithm with the number of morphemes in the SGSD (referred to as the "# Morphs" in the column title of the result tables). This gives a very shallow estimate of the performance of the algorithm because morpheme segmentation of a word can be incorrect even though the number of morphemes agrees. However, it provides an opportunity to estimate *over-segmentations* and *under-segmentations* of the algorithm useful for locating the source of errors.

5.2 Results

As described in section 4.1, we trained the Morfessor Categories-MAP algorithm using two different data sets and tested it using the data set described in section 4.2. We then obtained the results for each of the POS categories using the criteria explained in section 5.1. Table 5 shows the estimated accuracies (as a percentage) of each category for the Full List.

POS category	Exact	Stem	# Morph
Nouns	24.72	46.08	46.00
Verbs	26.89	33.42	60.96
Adjectives	32.29	32.32	32.36
Adverbs	12.42	18.15	19.75
Function Words	58.19	59.31	61.76
Overall	**25.56**	**42.06**	**47.45**

Table 5: Accuracy of the algorithm against the Full List

As can be expected, the accuracy of the algorithm increases when the evaluation moves from exact mappings to more shallow equivalences. According to the figures in Table 5, the algorithm performed better on predicting noun stems than verbs, even though it manages to segment more than 60% of the verbs into the correct number of morphemes. As shown in Table 4, in Sinhala, only nouns and verbs have more than two morphemes per lexeme. Therefore the accuracy of other categories does not vary too much among the three metrics of evaluation. Adverbs give the worst performance, which may be due to the lowest ratio of availability among other POS categories (see table 3).

Table 6 shows the estimated accuracies of the algorithm on the Restricted List. A significant improvement in the accuracy can be seen from the results in Table 6 when compared to the results in table 5. This implies that the behavior of the Morfessor algorithm with a large amount of data is not as good as with more accurate data. As shown in Table 2, the Full List contains 229, 070 more words than the Restricted List and the Full list is twice as large as the Restricted List. However, all the extra words present in the Full List have occurred only once in UCSC 10M words Sinhala corpus and many of them are either typos or very uncommon words. On the other hand, the Restricted List can be considered as more accurate list than the Full List. Based on these facts, it can be concluded that the Morfessor algorithm performs well with lower numbers of more accurate data than large numbers of erroneous data, as would be expected of it.

POS category	Exact	Stem	# Morph
Nouns	35.30	63.10	51.38
Verbs	29.15	37.76	56.38
Adjectives	69.36	69.39	69.39
Adverbs	22.93	28.66	33.44
Function Words	68.23	69.57	72.35
Overall	**35.05**	**56.37**	**51.38**

Table 6: Accuracy of the algorithm against the Restricted List

According to the figures in Table 6, the accuracy obtained by comparing stems (column 3 in Table 6) is higher than the accuracy obtained by comparing number of morphemes (column 4 in Table 6). It is vice versa in Table 5 and that is mainly due to the significant improvement of detecting noun stems with the Restricted List. This indicates the fact that the accurate data is more helpful for detecting noun stems than for segmenting it into a number of morphemes. The behavior of other POS categories is similar to the noun category except

for the fact that the accuracy of each category has improved proportionally. We obtained the number of morphemes output for each POS category as shown in Table 7.

POS Category	**#=1 (%)**	**#=2 (%)**	**#=3 (%)**	**#=4 (%)**	**#=5 (%)**
Nouns	13.3	38.2	33.2	12.8	2.7
Verbs	30.3	44.5	19.2	5.0	1.0
Adjectives	55.5	36.2	6.9	1.2	0.2
Adverbs	17.8	51.7	25.5	5.0	-
Function Words	54.4	34.5	10.0	1.1	-

Table 7: Number of Morphemes output for each POS category

By comparing the values of Table 5 and Table 6, it can be concluded that the algorithm has segmented nouns, verbs and adjectives fairly well, but has not done so well for adverbs and function words. The distributions of the number of morphemes of noun and adjective categories are similar for both test data and the output. We may conclude that the Morfessor algorithm performs well for nouns and adjectives while it encounters some problems in the segmentations of adverbs and verbs as seen by figures in table 5 and 6.

5.3 Error Analysis

We analyzed the miss-segmented words to identify the behavior of the algorithm against the Sinhala Language. We did the analysis only for the output obtained from the Restricted List since it gives better results than the Full List. We calculated the over-segmentation and the under-segmentation percentages for each category as shown in Table 8.

POS Category	**Correct-(-seg.)**	**Over-(-seg.)**	**Under-(-seg.)**
Nouns	51.38	30.58	18.75
Verbs	56.38	29.22	15.38
Adjectives	69.39	0.48	31.91
Adverbs	33.44	5.73	68.79
Function Words	72.35	4.24	23.63
Overall	**51.38**	**29.88**	**18.95**

Table 8: over-segmentation and under-segmentation percentages for each category

In Sinhala, a single word can be a noun, verb, adjective or an adverb, but its POS tag can vary based on its context. All appropriate POS categories of such words have been defined in SGSD. The evaluation algorithm was treated for such words in a way that if the segments of the output matched with any definition in SGSD, it will be treated as a successful segmentation. On the other hand, if none of definitions matched with the output, the counts for the error increases for each category. Therefore the sum of the rows in Table 8 can be higher than 100.

The figures in Table 8 prove that the most common drawback of the Morfessor algorithm for Sinhala is over-segmentation. However, even though over-segmentation is a drawback for nouns and verbs, the most common issue of other POS categories is under-segmentation. This may be due to lack of derivational forms of these categories in the training data. The accuracy of segmenting adverbs and adjectives has particularly suffered by this under-segmentation.

6 Conclusions

This work is carried out to evaluate how state-of-the-art machine learning algorithms for morph segmentation work with the Sinhala, an agglutinative language. No previous attempts have been reported in the literature, which analyze the behavior of the morphology of Indic languages such as Sinhala against a machine learning approach. This work confirms that the algorithm can obtain 35% linguistically exact morpheme analyses for Sinhala words. Furthermore, if such algorithms need to be used to build a stemmer for Sinhala, then we can expect more than 50% accuracy. This should be the base-line for future machine leaning approaches for Sinhala Morphology.

This work also shows that Sinhala nouns can be handled more accurately than Sinhala verbs using machine leaning approaches. As per Sinhala Gold Standard Definitions, Sinhala nouns can be inflected some 50 different forms while Sinhala verbs can be inflected to more than 200 forms. This makes the analysis of verbs more complicated than nouns. Segmenting other major POS categories seems to be more straightforward using machine learning algorithms since they are not so highly inflected.

7 Future Work

This work can be further extended by training the Morfessor algorithm for each POS category separately and testing for the same test list. It will help to identify Morfessor´s ability to adapt to the behavior of particular POS categories and also to identify the adaptability of machine learning for each POS.

Acknowledgment

The authors would like to acknowledge the National Research Council (NRC), Sri Lanka for funding this research. This work is supported in part by the LK Domain Registry. The conclusions and/or recommendations expressed however are those of the author and may not necessarily reflect the views of the LK Domain Registry. Also the authors are very grateful to past and current members at the Language Technology Research Laboratory of the University of Colombo of School Computing, Sri Lanka for their significant contribution on developing basic linguistic resources for Sinhala language such as corpora and lexica, which make it possible for us to carry out the research described above.

References

Jonathan Allen, M. Sharon Hunnicutt, and Dennis Klatt. 1987. *From Text to Speech: The MITalk System*. Cambridge.

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes.

Mathias Creutz and Krista Lagus. 2004. Induction of a simple morphology for highly-inflecting languages. In *In Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, pages 43–51.

Mathias Creutz and Krista Lagus. 2005a. Inducing the morphological lexicon of a natural language from unannotated text. In *In Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR05*, pages 106–113.

Mathias Creutz and Krista Lagus. 2005b. Unsupervised morpheme segmentation and morphology induction from text corpora using morfessor 1.0. In *Helsinki University of Technology*.

Mathias Creutz and Krista Lagus. 2006. Morfessor in the morpho challenge. In *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process*.

Mathias Creutz and Krista Lagus. 2012. Morfessor categories-map 0.9.2 download, April.

Herve Dejean. 1998. Morphemes as necessary concept for structures discovery from untagged corpora.

John Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. *Computational linguistics*, 27(2):153–198.

Mikko Kurimo, Sami Virpioja, Ville Turunen, and Krista Lagus. 2010. Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95, Uppsala, Sweden, July. Association for Computational Linguistics.

Mikko Kurimo, Krisra Lagus, Sami Virpioja, and Ville Turunen. 2011. Morpho challenge 2010 - semi-supervised and unsupervised analysis, September.

Ruvan Weerasinghe, Dulip Herath, Viraj Welgama, Nishantha Medagoda, Asanka Wasala, and Eranga Jayalatharachchi. 2007. Ucsc sinhala corpus - pan localization project-phase i.

Ruvan Weerasinghe, Dulip Herath, and Viraj Welgama. 2009. Corpus-based Sinhala lexicon. In *Proceedings of the 7th Workshop on Asian Language Resources*, pages 17–23, Suntec, Singapore, August. Association for Computational Linguistics.

A Transliteration Scheme used to Romanize Sinhala Script

Sinhala Script	Roman Script
◌ං	z
◌ඃ	Z
අ	a
ආ	A
ඇ	æ
ඈ	Æ
ඉ	i
ඊ	I
උ	u
ඌ	U
ඍ	R

ඍaa	H
ඵ	î
ඵෘ	Î
එ	e
ඒ	E
ඓ	X
ඔ	o
ඕ	O
ඖ	Y
ක	k
ඛ	K
ග	g
ඝ	G
ඞ	F
ඟ	β
ච	c
ඡ	C
ජ	j
ඣ	J
ඤ	ñ
ඥ	Ñ
ඦ	ç
ට	t
ඨ	T
ඩ	d
ඪ	D
ණ	N
ඬ	W
ත	q
ථ	Q
ද	v
ධ	V
න	n
ඳ	μ
ප	p
ඵ	P
බ	b
භ	B
ම	m

ඹ	M
ය	y
ර	r
ල	l
ව	w
ශ	S
ෂ	x
ස	s
හ	h
ළ	L
ෆ	f
්	
ා	A
ැ	æ
ෑ	Æ
ි	i
ී	I
ු	u
ූ	U
ෘa	R
ෘaa	H
ෘෟ	î
ෘෟ	Î
ෙ	e
ේ	E
ෙෙ	X
ො	o
ෙෝ	O
ෞ	Y