The Study of Effect of Length in Morphological Segmentation of Agglutinative Languages

Loganathan Ramasamy and **Zdeněk Žabokrtský**
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics, Charles University in Prague
{ramasamy, zabokrtsky}@ufal.mff.cuni.cz

Sowmya Vajjala
Seminar für Sprachwissenschaft
Universität Tübingen
sowmya@sfs.uni-tuebingen.de

Abstract

Morph length is one of the indicative feature that helps learning the morphology of languages, in particular agglutinative languages. In this paper, we introduce a simple unsupervised model for morphological segmentation and study how the knowledge of morph length affect the performance of the segmentation task under the Bayesian framework. The model is based on (Goldwater et al., 2006) unigram word segmentation model and assumes a simple prior distribution over morph length. We experiment this model on two highly related and agglutinative languages namely Tamil and Telugu, and compare our results with the state of the art Morfessor system. We show that, knowledge of morph length has a positive impact and provides competitive results in terms of overall performance.

1 Introduction

Most of the NLP tasks require one way or another the handling of morphology. The task becomes very crucial when the language in question is morphologically rich as is the case in many Indo-European languages. The application of morphology is evident in applications such as Statistical Machine Translation (SMT) (Lee, 2004), dependency parsing, information retrieval and so on. Apart from the morphological analysis as in the traditional linguistic sense, morphological segmentation is also widely used as an easy alternative to full fledged morphological analysis. In this paper we mainly focus on the task of morphological segmentation.

The main task in morphological segmentation is to segment the given *token* or *wordform* into set of morphs or identifying the location of each morpheme boundary within the *token*. Morphological segmentation is most suitable for agglutinative languages (such as Finnish or Turkish) than fusional languages (such as Semitic languages).

Though both supervised (Koskenniemi, 1983) and unsupervised methods (Goldsmith, 2001; Creutz and Lagus, 2005) are extensively studied for morphological segmentation, unsupervised techniques have the appeal of application to multilingual data with cost effective manner. Within unsupervised paradigm, various methods have been explored. Minimum Description Length (MDL) (Goldsmith, 2001; Creutz and Lagus, 2005) based approaches are most popular in which the best segmentation corresponds to the compact representation of morphology and the resulting lexicon. (Goldwater et al., 2009; Snyder and Barzilay, 2008) attempted word segmentation and joint segmentation of related languages using Bayesian approach. (Demberg, 2007; Dasgupta and Ng, 2007) applied various probabilistic measures to discover affixes of wordforms. (Naradowsky and Goldwater, 2009; Yarowsky and Wicentowski, 2000) explored ways to model orthographic rules of wordforms.

In this work, we are mainly going to focus on Bayesian approach. Bayesian approaches provide natural way of modeling subjective knowledge as well as separating problem specific aspects from general aspects. In the case of agglutinative lan-

Proceedings of the First Workshop on Multilingual Modeling, pages 18–24,
Jeju, Republic of Korea, 8-14 July 2012.

guages, the number of morphemes in a word as well as morph length play a major role in morphological process. The main rationale for this work is to study linguistic factors (mainly *morph length*), so that language specific priors can be applied over different languages. This will especially be useful when modeling resource poor languages (RPL) with little or no data, as well as building resources for RPL from resource rich languages (RRL).

Towards that objective, our main contribution in this work is, we introduce a simple unsupervised segmentation model based on Bayesian approach and we study the effect of morph length prior for two agglutinative languages.

2 Previous Work

In this section, we briefly survey earlier works that utilized the morph length information, then we provide basis for our unsupervised morphological segmentation model and finally we list some prior works on morphological analysis/segmentation of Telugu and Tamil.

Snover (2001) used an exponential like distribution for morph length that decreased over word length, thus favoring shorter morph lengths. Our work is directly related to (Creutz, 2003) as it made use of prior distributions on morph length and frequency of morphs under maximum a posteriori (MAP) framework. Gamma distribution was used as a prior distribution for morph length. The main difference between (Creutz, 2003) and our work is that, we are going to experiment different morph lengths under Bayesian framework.

Naradowsky (2011) introduced an exponential length penalty to prevent the model from under segmentation results. It also emphasized that avoiding length penalty seriously affected the model. (Poon et al. , 2009) indirectly specified about the morph length by restricting the number of morphemes per word.

In this work, we mainly rely on Goldwater (2009; 2006) which conducted an extensive study on the application of Bayesian approach to word segmentation in child-directed speech utterances. It included both unigram and bigram models (based on Hierarchical Dirichlet Processes) for word segmentation. Gibbs sampling was used to extract samples (utterances with word boundaries) from posterior distribution. We apply the unigram model (Goldwater et al., 2009) to morphological segmentation where the word boundaries in speech utterances correspond to morpheme boundaries in wordforms.

Before we describe unsupervised morphological segmentation model, we briefly survey the existing work on Telugu and Tamil morphological segmentation/analysis.

Rao et al. (2011) described in detail, the preparation of a linguistic database for Telugu morphological analysis, compiling 2800 morphological categories and reported a coverage of 95-97%. They followed a word and paradigm model, which was considered to be better suited for agglutinative languages. The issue of out-of-vocabulary words was handled better in the rule based approach by (Ganapathiraju and Levin, 2006). They describe a rule-based morphological analyzer *TelMore* for Telugu nouns and verbs.

Aksharbharathi et al. (2004) describes the development of a generic morphological analysis shell that uses dictionaries along with Finite State Transducers based feature structures, to perform the morphological analysis of a word. The feature structures were derived from the standard rules of the grammar in respective languages. This was tested with Hindi, Telugu, Tamil and Russian.

Kiranmai et al. (2010) describe a supervised morphological analyzer with support vector machines.

For Tamil, morphological segmentation is rarely studied. Most of the work is done for morphological analysis of wordforms. Most of the analyzers use rule based approaches. Dhanalakshmi et al. (2009) used sequence labeling approach to morphological analysis of wordforms.

3 Unsupervised Morphological Segmentation

Consider a wordform (w) of length n composed of characters from alphabet L_A,

$$w = c_1 c_2 c_3 ... c_n$$

The main objective is to identify the character positions where morpheme boundaries occur. The

model we describe here is similar to the *cache model* described in (Goldwater et al., 2006) for word segmentation. We apply the same model to identify morpheme boundaries. The model makes decision at every character position in the wordform for the entire corpus. The hypothesis probability that no morpheme boundary at position i in wordform w is calculated as follows,

$$P(w_i^-|h) = \frac{n_{m_a} + \alpha P_0(m_a)}{N_m + \alpha} \quad (1)$$

m_a is a substring or a morph in the wordform w which contains the character position position i. n_{m_a} refers to number of times the morph m_a occurs in the history of morph counts N_m. In the case of having a boundary at position i, we will have two morphs to consider, one morph (m_a) to the left of position i (including i), and another morph (m_b) starting after i. The probability of having a morpheme boundary at position i is calculated in the same way as Equation 1, but this time with two morphs,

$$P(w_i^+|h) = \frac{n_{m_a} + \alpha P_0(m_a)}{N_m + \alpha} \cdot \frac{n_{m_b} + I(m_a == m_b) + \alpha P_0(m_b)}{(N_m + 1) + \alpha} \quad (2)$$

$I(m_a == m_b)$ takes the value 1 if both morphs are same, otherwise the value is 0. Also note that the additional 1 (due to previous factor) in the denominator of the second part of the equation. In both the equations, P_0 is a base distribution which can be utilized to put a bias over certain hypotheses. In our case, the base distribution (P_0) mainly assigns probability distribution over morph length. Additional linguistic factors can also be modeled this way. α is a *concentration parameter* which can be used to control P_0. Overall, the model (in equation 1 and 2) uses only unigram morph counts.

Every character position (except the last position) in a given word is a potential candidate that can have a morpheme boundary. To determine whether they really have morpheme boundary or not, for every character position i in w, we calculate hypothesis probabilities b_i^+ (i.e. has a morpheme boundary) and b_i^- (has no morpheme boundary). Having calculated the hypothesis probabilities, we choose the hypothesis by using a weighted coin flip. In our problem, we have only two hypotheses: (i) a morpheme boundary and (ii) no morpheme boundary. If the new hypothesis is different from the character's previous status, then appropriate data structures are updated. This procedure is repeated for many number of iterations.

3.1 Modeling morpheme length

We encode our beliefs about morph length via base distribution P_0. We chose *Poisson* distribution for modeling the length of the morphs. Poisson distribution utilizing morph length is defined as $P(l,k) = \frac{l^k e^{-l}}{k!}$, where l is an expected length of the morph and when supplied k, it returns the probability density of a morph having length k. We define two base distributions based on morph length prior,

$$P_0^A(m) = p(l,k) = \frac{l^k e^{-l}}{k!} \quad (3)$$

$$P_0^B(m) = p(m)p(l,k) = \frac{n_m}{\mid l_m \mid} \frac{l^k e^{-l}}{k!} \quad (4)$$

$p(m)$ is probability of the morph itself. $\mid l_m \mid$ - total number of substrings of length equal to the length of morph m. Morfessor (Creutz and Lagus, 2005) uses Zipfian distribution for frequencies and *gamma length prior* for modeling the length of the morphs. Setting a particular expected morph length effectively puts a bias towards that particular *morph length (l)*. We experiment both our base distributions over different morph lengths.

3.2 Inferencing

Gibbs sampling (Gilks et al., 1996) uses iterative procedure to repeatedly draw value of a variable given the current state of all other variables in the model. In our case, drawing a value is equal to determining whether there is a boundary at the character position, thus obtaining individual morphemes. We iteratively segment the given corpus or list of words into morphological segments. The intuitive idea is that, when we sample enough number of times i.e. drawing morphological segments of words given history of segments of all other words,

the sampler converges to the posterior distribution of the morphological segments of the entire corpus. The Algorithm 1 gives a general outline of how the Gibbs sampling procedure is applied to morphological segmentation.

Algorithm 1: Basic Sampling Procedure

Data: words, model
Result: Segmented words
begin
  $RandSeg \longleftarrow InitializeSegments(words)$
  $Baseline \longleftarrow Evaluate(RandSeg)$
  $CurrSeg \longleftarrow RandSeg$
  $MorphCounts \longleftarrow GetCounts(CurrSeg)$
  for $i \in iterations$ **do**
    for $j \in size(words)$ **do**
      for $k \in length(words[j])$ **do**
        $b_k^- \longleftarrow Calculate(P(words[j]_k^-))$
        $b_k^+ \longleftarrow Calculate(P(words[j]_k^+))$
        if $HasNoBoundaryAt(k)$ **then**
          add boundary at k with
          probability $\frac{b_k^+}{b_k^- + b_k^+}$
          no change at k with probability
          $\frac{b_k^-}{b_k^- + b_k^+}$
        if $HasBoundaryAt(k)$ **then**
          remove boundary at k with
          probability $\frac{b_k^-}{b_k^- + b_k^+}$
          no change at k with probability
          $\frac{b_k^+}{b_k^- + b_k^+}$
        $UpdateCurrSeg(CurrSeg)$
        $AdjustMorphCounts(MorphCounts)$

We use *temperature (T)* settings (not shown in the algorithm) to make the sampling procedure converge faster. We use 10 values (from 0.1 to 1.0) for T and raise the probability values of hypotheses to $(\frac{1}{T})$. Also, we make the *collection rate* very small, so that only few and substantially different samples (or morphological segmentation of the entire corpus) are collected.

4 Experimental Setup

The experiments are carried out for the unigram segmentation model (*unsup-uni*) as described in Section 3 and Morfessor system (Creutz and Lagus, 2005). For both Tamil and Telugu, we perform the following experiments: (i) baseline (ii) *unsup-uni* with base distribution P_0^A (*unsup-uni-p0-len*) (iii) *unsup-uni* with base distribution P_0^B (*unsup-uni-p0-lex-len*) and (iv) with Morfessor. For each system, we add some knowledge about morph length (l) and report the accuracy.

The experiments (ii), (iii) and (iv) use additional dataset known as *extra-data*. *Extra-data* is an unannotated/unsegmented data which augments the *test data* while training the systems. As *test data* with gold segmentation is very small, we feel this step is necessary to make the evaluation credible. The following subsection describes the datasets in detail.

Baseline system corresponds to random segmentation. We evaluate *baseline* system for morph lengths 1 to 10. For each morph length (l) experiment, we change the probability of adding a boundary at each character position to be $(\frac{1}{l})$ except at $l = 1$ where the probability is 0.75.

Unsup-uni-p0-len experiment uses base distribution P_0^A (see Section 3.1). We conduct this experiment in 2 steps: (i) running the Gibbs sampler with the *extra-data* and (ii) use the parameters (including morph counts) from step (i) and run the Gibbs sampler on *test data*. We set the expected morph length (l) in the base distribution P_0^A every time we run the experiment for different morph length. For the step (i), the Gibbs sampler is run for 10000 iterations with different *concentration parameter* (α). We collect samples every 1000 iterations and we store the last sample as our model along with other parameters. For step (ii), we use the model from step (i) and run the Gibbs sampler on *test data*. We collect the final sample as our predicted segmentation of the *test data* and perform evaluation on the predicted segmentation. In *unsup-uni-p0-lex-len* experiment, we use the base distribution P_0^B (see Section 3.1). P_0^B includes morpheme probability apart from the length prior. Experiments for *unsup-uni-p0-lex-len* is carried out in the same way as that of *unsup-uni-p0-len*.

We use *gamma distribution* length prior for experiments with Morfessor. We train Morfessor on *extra-data* for morph lengths 1 to 10. We change the expected length in the gamma prior for each morph length experiment. Then we run the Morfessor on *test data* with same parameters created during the training.

We use *Precision (P), Recall (R)* and *F-score (F)*

Lang.	Words	Chars	Morphs	Avg. m.*(l)*
Tamil	1500	12642	3280	3.85
Telugu	998	10303	1733	5.95

Table 1: Gold segmentation: statistics

for evaluating our predicted segmentation with gold segmentation. Our evaluation is same as (Creutz and Lindén, 2004).

4.1 Data

We use EMILLE corpus (Xiao et al. , 2004) for our experiments. The EMILLE corpus contains monolingual, parallel and annotated data for various Indian languages. We randomly selected articles from *monolingual* section of Tamil and Telugu data. The original data were in `utf-8` and we transliterated the data into `latin` format. The transliteration step is an important step as it avoids confusion in specifying *morph length (l)*. As we already mentioned earlier, we use two sets (*extra-data* and *test data*) of data for each language. For training of *extra-data*, we use 30000 unique words list for each language. For *test data*, we make words list from real sentences thus it can contain multiple occurrences of a same wordform. The Table 1 provides the statistics of the *test data* for which we have manually performed gold segmentations. At present, our gold segmentation does not take into account multiple possible segmentations.

The Figure 1 shows morph counts distribution of both Tamil and Telugu (derived from gold segments) according to their morph lengths. Tamil has more morphs that are shorter in length than Telugu.

5 Results

The Table 2 shows evaluation results for the experimental setup described in the previous section.

For Tamil, most of the morphs have the length 1-4. The models *unsup-uni-p0-len* and *unsup-uni-p0-lex-len* perform quite well near to that length range. For the same range (l = 1 to 4), both the models together perform better than Morfessor in terms of F-score. The performance of *unsup-uni-p0-len* and *unsup-uni-p0-lex-len* are constantly decreasing and start to perform worse than Morfessor after length 5. This is somewhat expected that *unsup-uni* models are quite sensitive to length priors and may perform poorly if we assume morph lengths far from the true range. Whereas, Morfessor has a consistent performance over the entire length range (l = 1 to 10). This implies that, Morfessor is less sensitive to length priors even if we drastically change the expected morph length. *Unsup-uni-p0-len* gave the best overall performance (F-score - 48.83%) compared to other models in this task.

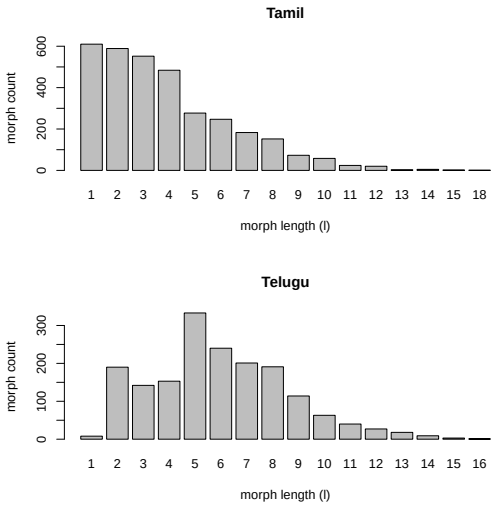

Figure 1: Morph counts according to morph length (l)

Telugu's common morph length ranges from 2-8. Except at $l = 1$ & 2, Morfessor beats both *unsup-uni-p0-len* and *unsup-uni-p0-lex-len* in all other remaining length ranges. *Unsup-uni* models perform quite poorly over different length ranges when comparing with Tamil for the same range. In this task, Morfessor's overall performance (F-score 43.63%) is better than *unsup-uni* models. Morfessor also performs better near the most frequent morph length range (5-8).

6 Some Observations on (*l*)

- The results (Table 2) suggest that *unsup-uni* model is quite sensitive to morph length parameter in the prior distributions.

- For Tamil, *unsup-uni* model performs well near to the true morph length range. But the performance deteriorates when the expected morph length parameter is too different from

Language	System	P/R/F	Morph length *(l)*									
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Tamil	baseline	P	15.79	15.86	17.04	17.11	15.33	16.33	15.98	14.75	17.63	16.65
		R	73.98	50.08	34.92	26.25	19.64	15.50	13.82	11.47	12.31	10.24
		F	26.02	24.09	22.91	20.72	17.22	15.91	14.82	12.91	14.50	12.68
	unsup-uni-p0-len	P	63.61	62.17	67.99	69.68	69.22	72.77	72.29	68.70	66.73	64.08
		R	39.62	40.01	36.49	33.18	28.82	26.47	24.23	22.10	20.65	20.76
		F	**48.83**	*48.69*	*47.49*	*44.96*	40.7	38.82	36.30	33.45	31.54	31.36
	unsup-uni-p0-lex-len	P	46.51	59.48	63.79	63.69	56.10	54.58	50.29	48.18	45.99	50.39
		R	41.35	41.07	39.34	38.28	36.04	33.69	34.25	34.08	33.02	28.65
		F	43.78	*48.59*	*48.67*	*47.82*	43.88	41.66	40.75	39.92	38.44	36.53
	Morfessor	P	48.54	48.32	48.61	49.01	50.24	49.07	49.93	49.21	49.42	48.93
		R	41.75	40.18	40.07	40.24	40.46	39.84	40.35	39.84	40.40	39.62
		F	44.89	43.87	43.93	44.19	44.82	43.98	44.63	44.03	44.64	43.78
Telugu	baseline	P	07.88	08.05	07.91	07.38	07.70	07.54	07.62	08.52	08.96	07.91
		R	75.69	51.59	32.97	23.86	20.00	16.00	13.66	13.38	12.97	10.07
		F	14.28	13.93	12.76	11.27	11.12	10.25	09.78	10.41	10.60	10.07
	unsup-uni-p0-len	P	36.67	37.29	36.2	39.71	41.87	40.58	41.34	39.15	38.10	33.65
		R	53.10	51.17	48.14	38.07	29.1	19.31	16.14	11.45	11.03	9.66
		F	43.38	43.14	41.33	38.87	34.34	26.17	23.21	17.72	17.11	15.01
	unsup-uni-p0-lex-len	P	22.27	26.55	32.46	35.76	28.29	19.31	19.83	18.3	18.17	17.26
		R	66.9	58.34	44.41	35.17	35.31	55.17	42.21	49.79	55.45	52.28
		F	33.41	36.5	37.51	35.47	31.41	28.6	26.98	26.76	27.37	25.95
	Morfessor	P	29.32	29.59	30.48	30.72	30.88	30.85	31.31	30.34	29.88	30.40
		R	70.30	69.48	69.48	69.75	70.17	70.30	71.96	70.99	70.58	71.96
		F	41.38	41.50	42.38	42.65	42.89	42.88	**43.63**	42.51	41.99	42.74

Table 2: Results for Tamil and Telugu

the true frequent morph length range.

- However for Telugu, morph length parameter did not improve the results at the most frequent morph length range (5-8).
- *Concentration parameter (α)* too influences the effect of base distribution as a whole, but at present, our study does not take into account α. For small α values, the base distribution will not have much effect.

7 Conclusion

In this paper, we mainly studied the effect of knowledge of morph length that could have on the accuracy of morphological segmentation of agglutinative languages. Towards that goal, we introduced a simple unsupervised morphological segmentation model based on Bayesian approach that utilized prior distribution over morph length. The results showed that the knowledge of length certainly has a positive impact on the accuracy. Also, the model provided competitive results in general and achieved best overall performance (F-score: 48.83%) for Tamil against Morfessor. As a future work, it would be interesting to see the model and priors that handle *sandhi* changes.

Acknowledgements

The research leading to these results has received funding from the European Commission's 7th Framework Program (FP7) under grant agreement n° 238405 (CLARA). We would like to thank David Mareček for useful suggestions about theory and implementation of the system. We also would like to thank anonymous reviewers for their useful comments.

References

Akshar Bharathi, Rajeev Sangal, Dipti M Sharma and Radhika Mamidi. 2004. Generic Morphological Analysis Shell. *In Proceedings of LREC 2004*.

Benjamin Snyder and Regina Barzilay. 2008. Unsupervised Multilingual Learning for Morphological Segmentation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 737–745. 2008.

David Yarowsky and Richard Wicentowski. Minimally Supervised Morphological Analysis by Multimodal Alignment. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (ACL)*, 2000.

Dhanalakshmi V, AnandKumar M, Rekha RU and Rajendran S. 2009. Morphological Analyzer for Ag-

glutinative Languages Using Machine Learning Approaches. In *Advances in Recent Technologies in Communication and Computing*, 2009, ARTCom'09, 2009.

Hoifung Poon, Colin Cherry and Kristina Toutanova. 2009. Unsupervised Morphological Segmentation with Log-Linear Models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL (NAACL-HLT)*, pages 209–217, Boulder, Colorado, June 2009.

Jason Naradowsky and Sharon Goldwater. 2009. Improving Morphology Induction by Learning Spelling Rules. In *Proceedings of 21st International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

Jason Naradowsky and Kristina Toutanova. 2011. Unsupervised Bilingual Morpheme Segmentation and Alignment with Context-rich Hidden Semi-Markov Models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 895–904, June, 2011.

John Goldsmith. 2001. Unsupervised Learning of the Morphology of a Natural Language. *Computational Linguistics*, 27(2): pages 153–198, 2001.

Kimmo Koskenniemi. 1983. Two-level morphology: A general computational model for word-form recognition and production. Publication 11, University of Helsinki, Department of General Linguistics, Helsinki. 1983.

Madhavi Ganapathiraju and Lori Levin. 2006. TelMore: Morphological Generator for Telugu Nouns and Verbs. *In Proceedings of the Second International Conference on Digital Libraries*. 2006.

Mathias Creutz. 2003. Unsupervised Segmentation of Words Using Prior Distributions of Morph Length and Frequency. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 280–287, July 2003.

Mathias Creutz and Krista Lagus. 2005. Unsupervised Morpheme Segmentation and Morphology Induction from Text Corpora Using Morfessor 1.0. In *Publications in Computer and Information Science*, Report A81, Helsinki University of Technology, 2005.

Mathias Creutz and Krister Lindén. 2004. Morpheme Segmentation Gold Standards for Finnish and English. Publications in Computer and Information Science, Report A77, Helsinki University of Technology, October, 2004.

Matthew G. Snover and Michael R. Brent. 2001. A Bayesian model for morpheme and paradigm identification. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics (ACL)*, pages 490–498, 2001.

Sai Kiranmai G., K. Mallika, M. Anand Kumar, V. Dhanalakshmi and K. P. Soman. 2010. Morphological Analyzer for Telugu using support vector machines. *In Proceedings of ICT 2010*.

Sajib Dasgupta and Vincent Ng. 2007. High-Performance, Language-Independent Morphological Segmentation. In *Proceedings of NAACL HLT 2007*, pages 155–163, 2007.

Sharon Goldwater, Thomas L. Griffiths and Mark Johnson. 2006. Contextual dependencies in unsupervised word segmentation. In *In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2006.

Sharon Goldwater, Thomas L. Griffiths and Mark Johnson. 2009. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112 (1), pp. 21–54, 2009.

Uma Maheshwar Rao G., Amba Kulkarni P. and Christopher Mala. 2011. A Telugu Morphological Analyzer. *International Telugu Internet Conference Proceedings, Milpitas, California, USA, 28th - 30th September, 2011*

Vera Demberg. 2007. A Language-Independent Unsupervised Model for Morphological Segmentation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 920–927, Prague, Czech Republic, June 2007.

Walter R. Gilks, Sylvia Richardson and David Spiegelhalter. 1996. Markov Chain Monte Carlo in Practice. Chapman and Hall. 1996.

Xiao Z., McEnery A., Baker P. and Hardie A. 2004. Developing Asian language corpora: standards and practice. In *Proceedings of the Fourth Workshop on Asian Language Resources*, pp. 1–8, 2004.

Young-Suk Lee. 2004. Morphological Analysis for Statistical Machine Translation. In Proceedings of the HLT-NAACL 2004, pp. 57–60, Boston, USA, 2004.