Statistical Morph Analyzer (SMA++) for Indian Languages

Saikrishna Srirampur
IIIT Hyderabad
saikrishna.srirampur
@research.iiit.ac.in

Ravi Chandibhamar
IIIT Hyderabad
chandibhamar.ravi
@students.iiit.ac.in

Radhika Mamidi
IIIT Hyderabad
radhika.mamidi
@iiit.ac.in

Abstract

Statistical morph analyzers have proved to be highly accurate while being comparatively easier to maintain than rule based approaches. Our morph analyzer (SMA++) is an improvement over the statistical morph analyzer (SMA) described in Malladi and Mannem (2013). SMA++ predicts the gender, number, person, case (GNPC) and the lemma (L) of a given token. We modified the SMA in Malladi and Mannem (2013), by adding some rich machine learning features. The feature set was chosen specifically to suit the characteristics of Indian Languages. In this paper we apply SMA++ to four Indian languages viz. Hindi, Urdu, Telugu and Tamil. Hindi and Urdu belong to the Indic[1] language family. Telugu and Tamil belong to the Dravidian[2] language family. We compare SMA++ with some state-of-art statistical morph analyzers viz. Morfette in Chrupała et al. (2008) and SMA in Malladi and Mannem (2013). In all four languages, our system performs better than the above mentioned state-of-art SMAs.

1 Introduction

Morphological analysis for Indian Languages (ILs) is defined as the analysis of a word in terms of its lemma (L), gender (G), number (N), person (P), case (C), vibhakti[3], tense, aspect and modality. A tool which predicts Morph Analysis of a word is called a Morph Analyzer (MA).

Statistical Morph Analyzer (SMA) is an MA which uses machine learning to predict the morph information . Using the training data and the feature-set, statistical models are formed. These models help to predict the morph-analysis of the test data. This works for all words, including out of vocabulary (OOV) words. SMA is language independent. We chose Indian Languages for our study and built an SMA which is targeted for different ILs.

Indian languages are lexically and grammatically similar. Lexical borrowing[4] occurs between languages. Gramatically, there are many similarities. Indian languages are *synthetic*[5]; derivational and inflectional morphologies result in the formation of complex words by stringing two or more morphemes. ILs predominantly have *subject-object-verb (SOV)* word order. They show agreement[6] among words. We captured such type of characteristics, by building a robust feature set.

2 Related Work

Traditionally, morphological analysis for Indian languages has been done using the rule based approach. For Hindi, the MA by Bharati et al. (1995) is most widely used among the NLP researchers in the Indian Community. Goyal and Lehal (2008) and Kanuparthi et al. (2012) MAs are advanced versions of the Bharati et al. (1995)'s analyzer. Kanuparthi et al. (2012) built a derivational MA for Hindi by introducing a layer over the Bharati et al. (1995)'s MA .It identifies 22 derivational suffixes which help in providing derivational analysis for the word whose suffix matches with one of these 22 suffixes.

[1] The Indic languages are the dominant language family of the Indian subcontinent, generally spoken in the regions of northern India and Pakistan

[2] The Dravidian languages are spoken mainly in southern India

[3] Vibhakti is a Sanskrit grammatical term that encompasses post-positionals and case endings for nouns, as well as inflection and auxiliaries for verbs. It is also referred as case-marker

[4] A word from one language that has been adapted for use in another is a borrowed word.

[5] a synthetic language is a language with a high morpheme-per-word ratio

[6] Agreement or Concord happens when a word changes form depending on the other words to which it relates

Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects, pages 103–109,
Dublin, Ireland, August 23 2014.

There have not been many updates in the rule based analyzers and the problem of not predicting OOV words is still a significant one. SMA in Malladi and Mannem (2013) is a data-driven MA which focuses primarily on Hindi.

For Urdu, Bögel et al. (2007) proposes an approach which uses Finite State Transducers. It introduces and discusses the issues that arise in the process of building finite-state MA for Urdu. For Telugu, Sunitha and Kalyani (2009) propose an approach of improving the existing rule based Telugu MA. They did this, using possible decompositions of the word, inflected by many morphemes. SMA in Malladi and Mannem (2013) evaluates the results for Urdu and Telugu as well. Not much research has been done in Morphological Analysis for Tamil.

3 Our Approach

3.1 Feature Set

The feature-set was chosen specifically to suit the Indian Languages. The following are the features used:

(i) Suffixes : Indian languages show inflectional morphology. The inflectional morphemes carry the G,N,P and C of a word. These morphemes generally occur in the form of suffixes. Hence, to capture the inflectional behaviour of ILs we considered the *suffixes* as a feature for the ML task. We considered suffixes whose length was maximum 7 characters.

(ii) Previous morph tags[7] and **next morph tags** : Agreement is an important characteristic of ILs. Through agreement, GNPC of a token may percolate to the other tokens. An example to this is, if the *subject* (noun) is masculine, then the verb form should also be masculine. To capture agreement, we considered features which carried the GNPC of the neighbouring words. *Previous morph tags* feature captures predicted morph tag of previous 3 tokens. *Next morph tags* feature captures the set of morph tags of the next token, if found in the training corpus.

(iii) Word Forms: ILs are morphologically rich languages. Words carry rich information regarding GNPC. To capture this characteristic we considered three features relating to word forms. *word_present* captures the word form of the present token. *word_previous* captures the word form of the previous token. *word_next* captures the word form of the next token.

(iv) Part of Speech (POS) : POS is one of the of the fundamental ML feature of any NLP task. Based on the POS of the word, the set of possible inflections can be found. For example, *verbs* have a set of inflections and *nouns* have another set. To capture such information we included POS in the feature-set.

(v) Other features : Features such as *length of the token* and *character types in the token* (eg. numbers, alphabets and so on) have also been considered.

The Support Vector Machine (SVM) (using linear classifier) was used for the ML task .

3.2 Choosing Class Labels

For the ML task, the class-labels for G, N, P, C were chosen from the training data itself. For lemma, the class-labels were formed based on the edit-distance[8] operations required to convert the given token to its lemma. This idea was inspired by Chrupała (2006), who introduced the concept of edit-operations[9] for lemmatization.

The Algorithm is explained using an example. Consider the token *crying*. The lemma for *crying* is *cry*.

Step 1: The token and its lemma are reversed. *crying* becomes *gniyrc* and *cry* becomes *yrc*.

Step 2: Note the edit operations required to convert reversed token to the reversed lemma. To convert *gniyrc* to *yrc* we need to delete the characters at the 1st, 2nd and 3rd indices. Hence the edit operations would be [d 1, d 2, d 3], where 'd' represents delete operation.

Step 3: The set of edit operations would form the class-label. [d 1, d 2, d 3] would be the class-label and would be added to the set of class-labels.

[7] The possible values of each G, N, P, C and L form the morph tags. eg. 'm' (masculine) is a morph tag for gender.

[8] Edit distance is a way of quantifying how dissimilar two strings (e.g., words) are to one another by counting the minimum number of operations required to transform one string into the other.

[9] The add, delete and replace operations required to convert one string to another

Similarly, the class-label for the token *playing* and the lemma *play* would be [d 1, d 2, d 3]. By this, *playing* - *play* and *crying* - *cry* have the same class label, because they have the common suffix *-ing*.

4 Experiments

Experiments were conducted for 4 ILs, viz. Hindi, Urdu, Telugu and Tamil. For Hindi, the Hindi Treebank (HTB) released as part of the 2012 Hindi Parsing Shared Task (Sharma et al., 2012) was used for the ML task. The statistical models were tuned on development data and evaluated on test data. Table 1. shows the HTB statistics.

For Urdu, the Urdu Treebank (UTB) released as a part of the 2012 Proceedings of TLT (Bhat and Sharma (2012)) was used for evaluation. Table 2. represents the UTB statistics. For Telugu, the Telugu Treebank (TTB) released for ICON 2010 Shared Task (Husain et al. (2010)) was used for evaluation. Table 3. represents the TTB statistics. For Tamil, the Tamil Treebank (TaTB) released by the The Indian Languages Machine Translation (ILMT)[10] project was used for evaluation. Table 4. represents the TaTB statistics.

Data	#Sentences	#Words
Training	12,041	268,096
Development	1,233	26,416
Test	1,828	39,775

Table 1: HTB Statistics.

Data	#Sentences	#Words
Training	5,700	159,743
Test	1,453	39,803

Table 2: UTB Statistics.

Data	#Sentences	#Words
Training	1300	5125
Test	150	600

Table 3: TTB Statistics.

[10] This consortium project is funded by Ministry of Communication and Information Technology, Technology Development for Indian Languages, Government Of India.

Data	**#Sentences**	**#Words**
Training	75	682
Test	25	271

Table 4: TaTB Statistics.

5 Results

The feature-set, which was specifically chosen for ILs, contributed to high accuracies. The results are shown for 4 Indian Languages. The results for each of L, G, N, P and C are shown individually, as well as in combination.

5.1 Hindi

The results are presented all five L ,G, N, P and C. The results are compared to 3 MAs viz. the traditional Rule Based MA (RBA) for Hindi, Morfette (M) in Chrupała et al. (2008) and SMA in Malladi and Mannem (2013) (SMA-M). There are two divisions for results. One for the Overall test data and other for the Out of Vocabulary (OOV) test data. SMA++ out performed other three MAs in almost all combinations. The results for OOV data are more pronounced. Table 5. shows the Hindi results.

Analysis	Test Data - Overall (%)				Test Data - OOV (%)			
	RBA	M	SMA-M	SMA++	RBA	M	SMA-M	SMA++
L	86.69	94.14	95.84	98.43	82.48	90.30	89.51	**93.07**
G	79.59	95.05	96.19	96.21	44.06	72.03	82.65	**83.11**
N	80.50	94.09	95.37	95.47	47.56	84.89	90.44	**92.81**
P	84.13	94.88	96.38	96.28	53.89	84.76	94.85	**96.17**
C	81.20	93.91	95.32	95.43	47.36	80.21	88.52	**89.45**
L+C	72.06	88.56	91.39	94.01	44.66	72.89	79.09	**82.92**
G+N+P	73.81	88.36	91.11	90.36	38.58	62.33	76.52	**77.24**
G+N+P+C	70.87	84.43	87.78	88.51	35.95	55.74	69.99	**72.36**
L+G+N+P	66.28	83.44	87.51	89.26	38.46	57.85	69.13	**72.82**
L+G+N+P+C	63.41	79.73	84.25	85.87	38.49	51.52	63.06	**65.96**

Table 5: Hindi Results

5.2 Urdu

The results are presented for L, G, N, P and C. The results are compared to 2 MAs viz. Morfette (M) in Chrupała et al. (2008) and SMA in Malladi and Mannem (2013) (SMA-M). Results are shown for both Overall test data and OOV test data. Even in Urdu, SMA++ out performed other two MAs in most of the combinations. Table 6. presents the results in comparison with Morfette (M) and Table 7. presents the results in comparison with SMA-M.

Analysis	Test Data - Overall (%)		Test Data - OOV (%)	
	M	SMA++	M	SMA++
L	93.65	95.34	87.54	**89.21**
G	90.39	93.79	79.40	**90.35**
N	92.38	95.66	85.36	**94.50**
P	93.93	97.07	86.56	**98.39**
C	87.99	90.92	76.08	**84.07**
L+C	82.94	86.93	67.25	**75.66**
G+N+P	84.52	89.43	70.32	**86.09**
G+N+P+C	77.01	82.17	58.54	**73.69**
L+G+N+P	80.12	86.07	64.14	**78.93**
L+G+N+P+C	73.11	79.16	53.30	**67.98**

Table 6: Urdu Results for SMA++ and M

Analysis	Test Data - Overall (%)		Test Data - OOV (%)	
	SMA-M	SMA++	SMA-M	SMA++
G	89.14	93.79	88.18	**90.35**
N	91.62	95.66	91.35	**94.50**
P	93.37	97.07	95.53	**98.39**
C	85.49	90.92	79.01	**84.07**

Table 7: Urdu Results for SMA++ and SMA-M

5.3 Telugu

The results are presented for G, N, P and C. The results are compared to 2 MAs viz. Morfette (M) in Chrupała et al. (2008) and SMA in Malladi and Mannem (2013) (SMA-M). Results are presented for both Overall test data and OOV test data. SMA++ significantly out performed Morfette (M). The results of *Overall Data* for SMA++ and SMA-M are very close, but more importantly the results of *OOV data* for SMA++ are higher than SMA-M. Table 8. presents the results in comparison with Morfette (M) and Table 9. presents the results in comparison with SMA-M.

Analysis	Test Data - Overall (%)		Test Data - OOV (%)	
	M	SMA++	M	SMA++
G	95.49	96.33	87.82	**89.85**
N	87.31	90.48	65.48	**77.67**
P	94.49	94.49	86.80	86.80
C	94.49	95.66	84.26	**90.36**
G+N+P	85.48	88.81	60.91	**74.62**
G+N+P+C	84.14	86.81	57.36	**70.56**

Table 8: Telugu Results for SMA++ and M

Analysis	Test Data - Overall (%)		Test Data - OOV (%)	
	SMA-M	SMA++	SMA-M	SMA++
G	96.49	96.33	89.85	89.85
N	90.65	90.48	75.13	**77.67**
P	94.82	94.49	85.79	**86.80**
C	96.49	95.66	89.34	**90.36**

Table 9: Telugu Results for SMA++ and SMA-M

5.4 Tamil

The results are presented for G, N, P and C. The results are compared to Morfette (M) in Chrupała et al. (2008). SMA++ out performs Morfette (M). Table 10. presents the results in comparison with Morfette (M).

Analysis	Test Data - Overall (%)		Test Data - OOV (%)	
	M	SMA++	M	SMA++
G	90.40	91.14	85.18	**91.36**
N	88.93	90.04	83.95	**87.04**
P	98.15	98.89	96.91	**98.14**
C	87.82	94.46	80.86	**91.98**
G+N+P	80.81	82.66	70.99	**80.25**
G+N+P+C	76.38	78.97	64.20	**74.07**

Table 10: Tamil Results

6 Conclusions and Future Work:

For all the four ILs, SMA++ out performs other SMAs. For Hindi, the L+G+N+P+C accuracy was **85.87%**. For Urdu, the L+G+N+P+C accuracy was **79.16%**. For Telugu, G+N+P+C accuracy was **86.81%** and for Tamil it was **78.97%**. These high values show that SMA++ is a marked improvement over the SMA in Malladi and Mannem (2013) . We studied two families of ILs, viz. Indic and Dravidian, because most of the ILs fall into these two groups. We plan to run SMA++ to predict Lemma in Telugu and Tamil. We plan to extend our work to European Languages such as Polish, German, French etc. We are currently working on the error analysis of our system. In future, we plan to deploy SMA++ for the ILMT project.

References

Akshar Bharati, Vineet Chaitanya, Rajeev Sangal, and KV Ramakrishnamacharyulu. 1995. *Natural language processing: a Paninian perspective*. Prentice-Hall of India New Delhi.

Riyaz Ahmad Bhat and Dipti Misra Sharma. 2012. A dependency treebank of urdu and its evaluation. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 157–165. Association for Computational Linguistics.

Tina Bögel, Miriam Butt, Annette Hautli, and Sebastian Sulger. 2007. Developing a finite-state morphological analyzer for urdu and hindi. *Finite State Methods and Natural Language Processing*, page 86.

Grzegorz Chrupała, Georgiana Dinu, and Josef Van Genabith. 2008. Learning morphology with morfette.

Grzegorz Chrupała. 2006. Simple data-driven contextsensitive lemmatization. *Procesamiento del Lenguaje Natural*, 37:121–127.

Vishal Goyal and Gurpreet Singh Lehal. 2008. Hindi morphological analyzer and generator. In *Emerging Trends in Engineering and Technology, 2008. ICETET'08. First International Conference on*, pages 1156–1159. IEEE.

Samar Husain, Prashanth Mannem, Bharat Ram Ambati, and Phani Gadde. 2010. The icon-2010 tools contest on indian language dependency parsing. *Proceedings of ICON-2010 Tools Contest on Indian Language Dependency Parsing, ICON*, 10:1–8.

Nikhil Kanuparthi, Abhilash Inumella, and Dipti Misra Sharma. 2012. Hindi derivational morphological analyzer. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pages 10–16. Association for Computational Linguistics.

Deepak Kumar Malladi and Prashanth Mannem. 2013. Context based statistical morphological analyzer and its effect on hindi dependency parsing. In *Fourth Workshop on Statistical Parsing of Morphologically Rich Languages*, volume 12, page 119.

Dipti Misra Sharma, Prashanth Mannem, Joseph vanGenabith, Sobha Lalitha Devi, Radhika Mamidi, and Ranjani Parthasarathi, editors. 2012. *Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages*. The COLING 2012 Organizing Committee, Mumbai, India, December.

KVN Sunitha and N Kalyani. 2009. A novel approach to improve rule based telugu morphological analyzer. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pages 1649–1652. IEEE.