Verbs are where all the action lies: Experiences of Shallow Parsing of a Morphologically Rich Language

Harshada Gune **Mugdha Bapat** **Mitesh M. Khapra** **Pushpak Bhattacharyya**
Department of Computer Science and Engineering,
Indian Institute of Technology Bombay
{harshadag,mbapat,miteshk,pb}@cse.iitb.ac.in

Abstract

Verb suffixes and verb complexes of morphologically rich languages carry a lot of information. We show that this information if harnessed for the task of shallow parsing can lead to dramatic improvements in accuracy for a morphologically rich language- Marathi[1]. The crux of the approach is to use a powerful morphological analyzer backed by a high coverage lexicon to generate rich features for a CRF based sequence classifier. Accuracy figures of 94% for Part of Speech Tagging and 97% for Chunking using a modestly sized corpus (20K words) vindicate our claim that for morphologically rich languages linguistic insight can obviate the need for large amount of annotated corpora.

1 Introduction

Shallow parsing which involves Part-of-Speech (POS) tagging and Chunking is a fundamental task of Natural Language Processing (NLP). It is natural to view each of these sub-tasks as a sequence labeling task of assigning POS/chunk labels to a given word sequence. For languages like English where annotated corpora are available in abundance these tasks can be performed with very high accuracy using data-driven machine learning techniques. Languages of the world show different levels of readiness with respect to such annotated resources and hence not all languages may provide a conducive platform for machine learning techniques.

In this scenario, morphologically rich languages from the Indian subcontinent present a very interesting case. While these languages do not enjoy the resource abundance of English, their linguistic richness can be used to offset this resource deficit. Specifically, in such languages, the suffixes carry a lot of information about the category of a word which can be harnessed for shallow parsing. This is especially true in the case of verbs where suffixes like णे {ne}, णारे {naare} [2] clearly indicate the category of the word. Further, the structure of verb groups in such languages is relatively rigid and can be used to reduce the ambiguity between main verbs and auxiliary verbs.

In the current work, we aim to reduce the data requirement of machine learning techniques by appropriate feature engineering based on the characteristics of the language. Specifically, we target Marathi- a morphologically rich language- and show that a powerful morphological analyzer backed by a high coverage lexicon and a simple but accurate Verb Group Identifier (VGI) can go a long way in improving the accuracy of a state of the art sequence classifier. Further, we show that harnessing such features is the only way by which one can hope to build a high-accuracy classifier for such languages, and that simply throwing in a large amount of annotated corpora does not serve the purpose. Hence it makes more sense to invest time and money in developing good morphological analyzers for such languages than investing in annotation. Accuracy figures of 94% for Part of

[1] Marathi is the official language of Maharashtra, a state in Western India. The language has close to 20 million speakers in the world.

[2] These are the suffixes which derive infinitive and gerund verb forms respectively.

Coling 2008: Poster Volume, pages 347–355,
Beijing, August 2010

Speech Tagging and 97% for Chunking using a modestly sized corpus (20K words) vindicate our claim that for morphologically rich languages linguistic knowledge plays a very important role in shallow parsing of these languages.

2 Related Work

Many POS taggers have been built for English employing machine learning techniques ranging from Decision Trees (Black et al., 1992) to Graphical Models (Brants, 2000; Brill, 1995; Ratnaparkhi, 1996; Lafferty et al., 2001). Even hybrid taggers such as CLAWS (Garside and Smith, 1997) which combine stochastic and rule based approaches have been developed. However, most of these techniques do not focus on harnessing the morphology; instead they rely on the abundance of data which is not a very suitable proposition for some of the resource deprived languages of the Indian sub-continent.

Morphological processing based taggers using a combination of hand-crafted rules and annotated corpora have been tried for Turkish (Oflazer and Kuruöz, 1994), Arabic (Tlili-Guiassa, 2006), Hungarian (Megyesi, 1999) and Modern Greek (Giorgos et al., 1999). The work on Hindi POS tagging (Singh et al., 2006) comes closest to our approach which showed that using a detailed linguistic analysis of morphosyntactic phenomena, followed by leveraging suffix information and accurate verb group identification can help to build a high-accuracy (93-94%) part of speech tagger for Hindi. However, to the best of our knowledge, there is no POS tagger and Chunker available for Marathi and ours is the first attempt at building one.

3 Motivating Examples

To explain the importance of suffix information for shallow parsing we present two motivating examples. First, consider the following Marathi sentence,

हा रस्ता दोन गावांना जोडणारा आहे.
*haa rasta don gavaannaa **jodaNaaraa_VM** aahe.*
*this road two villages **connecting** is*
*this is the road **connecting_VM** two villages.*

The word जोडणारा {jodaNaaraa} (connecting) in the above sentence is a verb and can be categorized as such by simply looking at the suffix णारा {Naaraa} as this suffix does not appear with any other POS category. When suffix information is used as a feature a statistical POS tagger is able to identify the correct POS tag of जोडणारा {jodaNaaraa} even when it does not appear in the training data. Hence, using suffix information ensures that a classifier is able to learn meaningful patterns even in the absence of large training data. Next, we consider two examples for chunking.

- **VGNN (Gerund Verb Chunk)**
  माणसाने उडण्याचा प्रयत्न केला.
  *maaNasaane **uDaNyaachaa_B-VGNN**[3]*
  prayatna kelaa.
  *man **fly** try do*
  *man tried **flying_B-VGNN**.*

- **VGINF (Infinitival Verb Chunk)**
  त्याने चालायला सुरुवात केली.
  *tyaane **chaalaayalaa_B-VGNF** suruvaata kelii.*
  *he **walk** start did*
  *he started **to walk_B-VGINF**.*

Here, we are dealing with the case of two specific verb chunks, *viz.*, VGNN (gerund verb chunk) and VGINF (infinitival verb chunk). A chunk having a gerund always gets annotated as VGNN and a chunk having an infinitival verb always gets annotated as VGINF. Thus, the correct identification of these verb chunks boils down to the correct identification of gerunds and infinitival verb forms in the sentence which in turn depend on the careful analysis of suffix information. For example, in Marathi, the attachment of the verbal suffix "ण्याचा" {Nyaachaa} to a verb root always results in a gerund. Similarly, the attachment of the verbal suffix "यला" {yalaa} to a verb root always results in an infinitival verb form. The use of such suffix information as features can thus lead to better generalization for handling unseen words and thereby reduce the need for additional training data. For instance, in the first sentence, even when the word "उडण्याचा" {uDaNyaachaa} does not appear in

[3]Note that for all our experiments we used BI scheme for chunking as opposed to the BIO scheme

the training data, a classifier which uses suffix information is able to label it correctly based on its experience of previous words having suffix "ण्याचा" {Nyaachaa} whereas a classifier which does not use suffix information fails to classify it correctly.

4 Morphological Structure of Marathi

Marathi nouns inflect for number and case. They may undergo derivation on the attachment of postpositions. In the oblique case, first a stem is obtained from the root by applying the rules of inflection. Then a postposition is attached to the stem. Postpositions (including case markers and the derivational suffixes) play a very important role in Marathi morphology due to the complex morphotactics.

Marathi adjectives can be classified into two categories: ones that do not inflect and others that inflect for gender, number and case where such an inflection agrees with the gender and number of the noun modified by them.

The verbs inflect for gender, number and person of the subject and the direct object in a sentence. They also inflect for tense and aspect of the action as well as mood of the speaker in an illocutionary act. They may even undergo derivation to derive the nouns, adjectives or postpositions. Verbal morphology in Marathi is based on *Aakhyaata* theory for inflection and *Krudanta* theory for derivation which are two types of verb suffixes (Damale, 1970).

Aakhyaata Theory: *Aakhyaata* refers to tense, aspect and mood. Aakhyaata form is realized through an aakhyaata suffix which is a closing suffix attached to verb root. For example, बसला {basalaa} (*sat*) comes from *basa* + *laa*. There are 8 types of aakhyaatas named after the phonemic shape of the aakhyaata suffix. Associated with every aakhyaata are various aakhyaata-arthas which indicate the features: tense, aspect and mood. An aakhyaata may or may not agree with gender.

Krudanta Theory: Krudanta suffixes are attached to the end of verbs to form non-infinitive verb forms. For example, धावायला (धाव + आयला) {dhaavaayalaa} (to run). There are 8 types of krudantas defined in Marathi.

5 Design of Marathi Shallow Parser

Figure 1 and 2 show the overall architectures of Marathi POS tagger and chunker. The proposed system contains 3 important components. First, a morphological analyzer which provides ambiguity schemes and suffix information for generating a rich set of features. Ambiguity Scheme refers to the list of possible POS categories a word can take. This can add valuable information to a sequence classifier by restricting the set of possible POS categories for a word. For example, the word जात {jaat} meaning caste or go(caste-noun, go- VM/VAUX) can appear as a noun or a main verb or an auxiliary verb. Hence it falls in the ambiguity scheme <NN-VM-VAUX>. This information is stored in a lexicon. These features are then fed to a CRF based engine which couples them with other elementary features (previous/next words and bigram tags) for training a sequence labeler. Finally, in the case of POS tagger, we use a Verb Group Identifier (VGI) which acts as an error correcting module for correcting the output of the CRF based sequence labeler. Each of these components is described in detail in the following sub-sections.

5.1 Morphological Analyzer

The formation of polymorphemic words leads to complexities which need to be handled during the analysis process. For example, consider the steps involved in the formation of the word देवासमोरच्याने {devasamorchyane} (the one in front of the God + ERGATIVE).

$$
\begin{aligned}
\text{devaasamora} &= (\text{deva} \rightarrow \text{devaa}) \\
&\quad + \text{samora} \\
\text{devaasamorachaa} &= (\text{devaasamora} \rightarrow \text{devaasamora}) \\
&\quad + \text{chaa} \\
\text{devaasamorachyaane} &= (\text{devaasamorachaa} \rightarrow \text{devaasamorachyaa}) \\
&\quad + \text{ne}
\end{aligned}
$$

In theory, the process can continue recursively for the attachment of any number of suffixes. However, in practice, we have observed that a word in Marathi contains at most 4 suffixes.

FSMs prove to be elegant and computationally efficient tools for analyzing polymorphemic

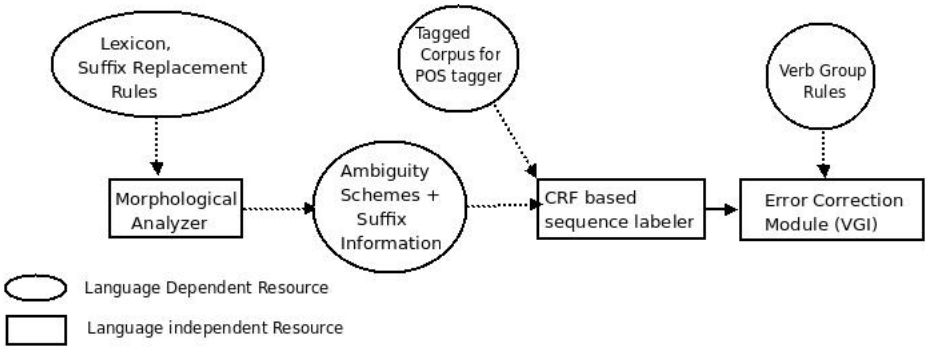

Figure 1: Architecture of POS Tagger

words. However, the recursive process of word formation in Marathi involves inflection at the time of attachment of every new suffix. The FSM needs to be modified to handle this. However, during the *i-th* recursion only *(i-1)-th* morpheme changes its form which can be handled by suitably modifying the FSM. The formation of word देवासमोरच्याने {devaasamorachyaane} can be viewed as:

devaasamora	=	(deva → devaa)
		+ samora
devaasamorachaa	=	(deva → devaa)
		+ (samora → samora)
		+ chaa
devaasamorachyaane	=	(deva → devaa)
		+ (samora → samora)
		+ (chaa → chyaa)
		+ ne

In general,
Polymorphemic word = $(inflected\_morpheme_1)$ + $(inflected\_morpheme_2)$ + ...

Now, we can create an FSM which is aware of these inflected forms of morphemes in addition to the actual morphemes to handle the above recursive process of word formation. These inflected forms are generated using the paradigm-based[4] system written in Java and then fed to the FSM implemented using SFST[5].

[4]A paradigm identifies the uninflected form of words which share similar inflectional patterns.

[5]http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/SFST.html

Our lexicon contains 16448 nouns categorized into 76 paradigms, 8516 adjectives classified as inflecting and non-inflecting adjectives, 1160 verbs classified into 22 classes. It contains 142 postpositions, 80 aakhyaata and 8 krudanta suffixes.

5.2 CRF

Conditional Random Fields (Lafferty et al., 2001) are undirected graphical models used for labeling sequential data. Under this model, the conditional probability distribution of a tag given the observed word sequence is given by,

$$P(Y|X;\lambda) = \frac{1}{Z(X)} \cdot e^{\sum_{t=1}^{T}\sum_{k=1}^{K}\lambda_k f_k(Y_{t-1},Y_t,X,t)} \quad (1)$$

where,

$$X = source\ word$$
$$Y = target\ word$$
$$T = length\ of\ sentence$$
$$K = number\ of\ features$$
$$\lambda_k = feature\ weight$$
$$Z(X) = normalization\ constant$$

We used CRF++[6], an open source implementation of CRF, for training and further decoding the tag sequence. We used the following features for training the sequence labeler (here, w_i is the *i-th* word, t_i is the *i-th* pos tag and c_i is the *i-th* chunk tag).

[6]http://crfpp.sourceforge.net/

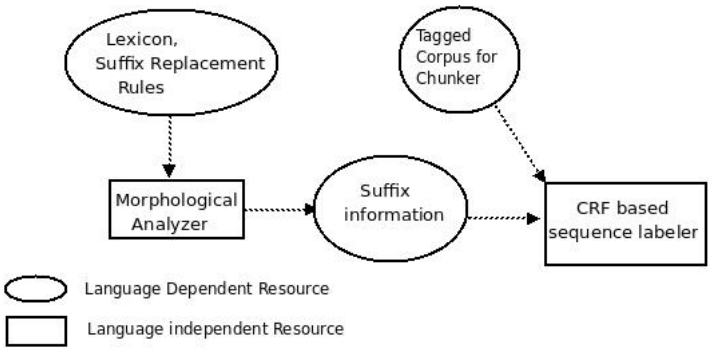

Figure 2: Architecture of Chunker

Features used for POS tagger training
Consider position of interest = i

- $t_i\ t_{i-1}$ and w_j such that $i-3 < j < i+3$
- $t_i\ t_{i-1}$ and suffix information of w_i
- $t_i\ t_{i-1}$ and ambiguity scheme of w_i

Here, the first features are ***weak features*** which depend only on the previous/next words and bigram tags. The next two are ***rich morphological features*** which make use of the output of the morphological analyzer.

Features used for Chunker training
Consider position of interest = i

- $c_i\ c_{i-1}$ and t_j, w_j such that $i-3 < j < i+3$
- $c_i\ c_{i-1}$ and suffix information of w_i

where c_i, $c_{i-1} \in \{B, I\}$. Here again, the first set of features are ***weak features*** and the second set of features are ***rich morphological features***.

5.3 Verb Group Identification (VGI)

In Marathi, certain auxiliaries like असते {asate} (be), आहे {aahe} (is) *etc.*. can also act as main verbs in certain contexts. This ambiguity between VM (main verbs) and VAUX (auxiliary verbs) can lead to a large number of errors in POS tagging if not handled correctly. However, the relatively rigid structure of Marathi VG coupled with distinct suffix-affinity of auxiliary verbs allows us to capture this ambiguity well using the following simple regular expression:

MainVerbRoot (KrudantaSuffix AuxVerbRoot)* AakhyaataSuffix

The above regular expression imposes some restriction on the occurrence of certain auxiliary verbs after specific *krudanta* suffixes. This restriction is captured with the help of a rule file containing *krudanta suffix-auxiliary verb* pairs. A sample entry from this file is

ऊन , काढ *[oon, kaaDh]*

which suggests that the auxiliary verb काढ {kaaDh} can appear after the suffix ऊन {oon}. We created a rule file containing around 350 such valid *krudanta suffix-auxiliary* verb pairs.

An important point which needs to be highlighted here is that a simple left to right scan ignoring suffix information and marking the first verb constituent as main verb and every other constituent as auxiliary verb does not work for Marathi. For example, consider the following verb sequence,

त्याला उचलून आणावे लागले.
tyaalaa uchaluun aaNaave laagale
He carry bring need
It was needed to carry and bring him.

Here, a simple left to right scan of the verb sequence ignoring the suffix information would imply that उचलून is a VM whereas आणावे and लागले are VAUX. However, this is not the case and can be identified correctly by considering the suffix affinity of auxiliary verbs. Specifically, in this case, the verb root आण cannot take the role of an auxiliary verb when it appears after the *krudanta* suffix ऊन. This suggests that the verb

आणावे does not belong to the same verb group as उचलून and hence is not a VAUX. This shows suffix and regular expression help in disambiguating VM-VAUX which is a challenge in all POS taggers.

6 Experimental Setup

We used documents from the TOURISM and NEWS domain for all our experiments [7]. These documents were hand annotated by two Marathi lexicographers. The total size of the corpus was kept large (106273 POS tagged words and 63033 chunks) to study the impact of the size of training data versus the amount of linguistic information used. The statistics about each POS tag and chunk tag are summarized in Table 1 and Table 2.

POS Tag	Frequency in Corpus	POS Tag	Frequency in Corpus
NN	51047	RP	359
NST	578	CC	3735
PRP	8770	QW	630
DEM	3241	QF	1928
VM	17716	QC	2787
VAUX	6295	QO	277
JJ	7311	INTF	158
RB	1060	INJ	22
UT	97	RDP	39
PSP	69	NEG	154

Table 1: POS Tags in Training Data

Chunk Tag	Frequency in Corpus	Chunk Tag	Frequency in Corpus
NP	40254	JJP	2680
VGF	7425	VGNF	3553
VGNN	1105	VGINF	58
RBP	782	BLK	2337
CCP	4796	NEGP	43

Table 2: Chunk Tags in Training Data

7 Results

We report results in four different settings:
Weak Features (WF): Here we use the basic CRF classifier with elementary word features (*i.e.*, words appearing in a context window of 3) and bigram tag features and POS tags in case of chunker.
Weak Morphological Features (Weak-MF): In addition to the elementary features we use substrings of length 1 to 7 appearing at the end of the word as feature. The idea here is that such substrings taken from the end of the word can provide a good approximation of the actual suffix of the word. Such substrings thus provide a statistical approximation of the suffixes in the absence of a full fledged morphological analyzer. This should not be confused with weak features which mean tags and word.
Rich Morphological Features (Rich-MF): In addition to the elementary features we use the ambiguity schemes and suffix information provided by the morphological analyzer.
Reach Morphological Features + Verb Group Identification (Rich-MF+VGI): This setting is applicable only for POS tagging where we apply an error correcting VGI module to correct the output of the feature rich CRF tagger.

In each case we first divided the data into four folds (75% for training and 25% for testing). Next, we varied the training data in increments of 10K and calculated the accuracy of each of the above models. The x-axis represents the size of the training data and the y-axis represents the precision of the tagger/chunker. Figure 3 plots the average precision of the POS tagger across all categories using WF, Weak-MF, Rich-MF and Rich-MF_VGI for varying sizes of the training data. Figure 6 plots the average precision of the chunker across all categories using WF, Weak-MF and Rich-MF. Next, to show that the impact of morphological analysis is felt more for verbs than other POS categories we plot the accuracies of verb pos tags (Figure 4) and verb chunk tags (Figure 7) using WF, Weak-MF, Rich-MF and Rich-MF_VGI for varying sizes of the training data.

8 Discussions

We made the following interesting observations from the above graphs and tables.
1. Importance of linguistic knowledge: Figure 3 shows that using a large amount of annotated corpus (91k), the best accuracy one can hope

[7] The data can be found at www.cfilt.iitb.ac.in/

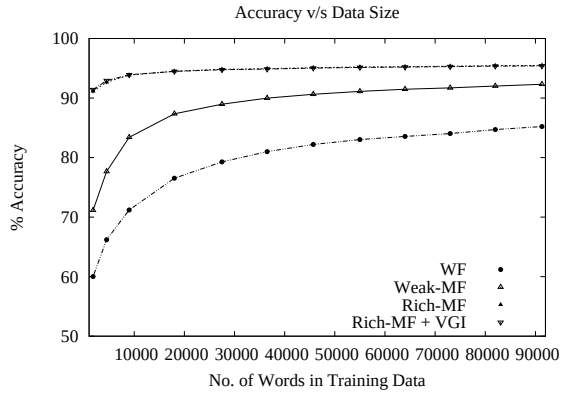

Figure 3: Average Accuracy of all POS Tags
(Note: The graphs for Rich-MF and Rich-MF+VGI coincide)

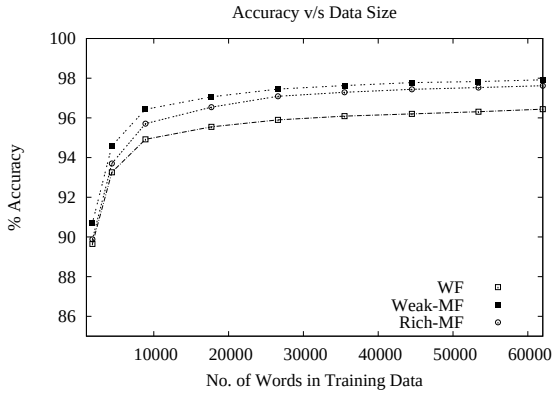

Figure 6: Average Accuracy of all Chunk Tags

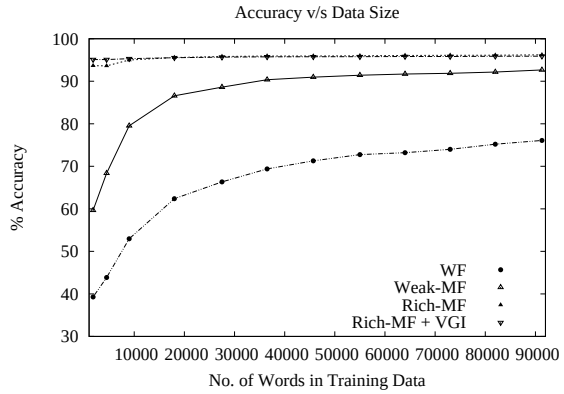

Figure 4: Average Accuracy of Verb POS Tags
(Note: The graphs for Rich-MF and Rich-MF+VGI almost coincide)

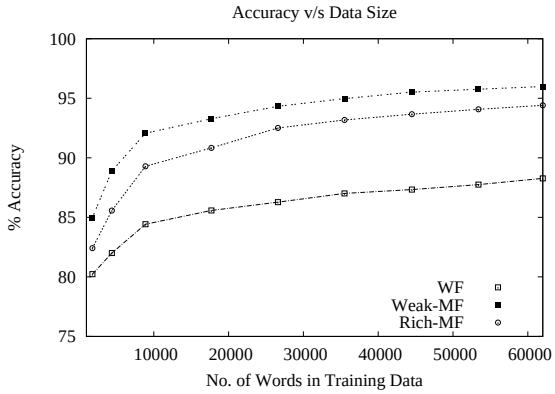

Figure 7: Average Accuracy of Verb Chunks

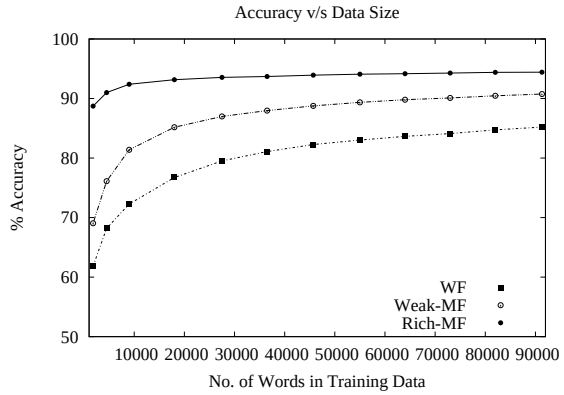

Figure 5: Average Accuracy of Non Verb POS Tags

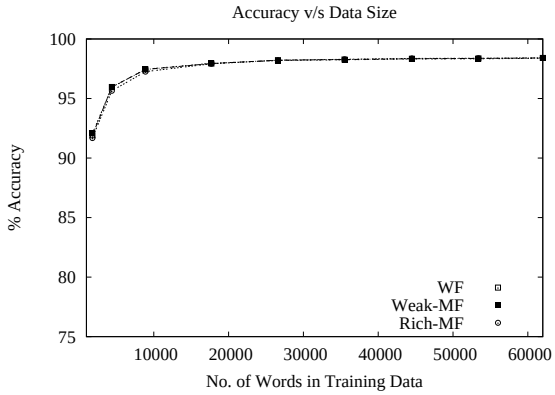

Figure 8: Average Accuracy of Non Verb Chunks
(Note: All the graphs coincide.)

for is around 85% if morphological information is not harnessed *i.e.*, if only weak features are used. Adding more data will definitely not be of much use as the curve is already close to saturation. On the other hand, if morphological information is completely harnessed using a rich morphological analyzer then an accuracy as high as 94% can be obtained by using data as small as 20k words. Figure 6 tells a similar story. In the absence of morphological features a large amount of annotated corpus (62k words) is needed to reach an accuracy of 96%, whereas if suffix information is used then the same accuracy can be reached using a much smaller training corpus (20k words). This clearly shows that while dealing with morphologically rich languages, time and effort should be invested in building powerful morphological analyzer.

2. Weak morphological features vs rich morphological analyzer: Figure 3 shows that in the case of POS tagging using just weak morphological features gives much better results than the baseline (i.e. using only weak features). However, it does not do as well as the rich features especially when the training size is small, thereby suggesting that an approximation of the morphological suffixes may not work for a language having rich and diverse morphology. On the other hand, in the case of chunking, the weak morphological features do marginally better than the rich morphological features suggesting that for a relatively easier task (chunking as compared to POS tagging) even a simple approximation of the actual suffixes may deliver the goods.

3. Specific case of verbs: Figure 4 shows that in case of POS tagging using suffixes as features results in a significant increase in accuracy of verbs. Specifically accuracy increases from 62% to 95% using a very small amount of annotated corpus (20K words). Comparing this with figure 5 we see that while using morphological information definitely helps other POS categories, the impact is not as high as that felt for verbs. Figures 7 and 8 for chunking show a similar pattern *i.e.*, the accuracy of verb chunks is affected more by morphology as compared to other chunk tags. These figures support our claim that "verbs is where all the action lies" and they indeed need special treatment in terms of morphological analysis.

	VM	VAUX
VM	17078	347
VAUX	257	6025

Table 3: Confusion matrix for VM-VAUX using Rich-MF

4. Effect of VGI: Figures 3 and 4 show that the VGI module does not lead to any improvement in the overall accuracy. A detailed analysis showed that this is mainly because there was not much VM-VAUX ambiguity left after applying CRF model containing rich morphological features. To further illustrate our point we present the confusion matrix (see Table 3) for verb tags for a POS tagger using Rich-MF. Table 3 shows that there were only 347 VM tags which got wrongly tagged as VAUX and 257 VAUX tags which got wrongly tagged as VM. Thus the rich morphological features were able to take care of most VM-VAUX ambiguities in the data. However we feel that if the data contains several VM-VAUX ambiguities such as the one illustrated in the example in Section 5.3 then the VGI module would come in play and help to boost the performance by resolving such ambiguities.

9 Conclusion

We presented here our work on shallow parsing of a morphologically rich language- Marathi. Our results show that while dealing with such languages one cannot ignore the importance of harnessing morphological features. This is especially true for verbs where improvements upto 50% in accuracy can be obtained by adroit handling of suffixes and accurate verb group identification. An important conclusion that can be drawn from our work is that while dealing with morphologically rich languages it makes sense to invest time and money in developing powerful morphological analyzers than placing all the bets on annotating data.

References

Black, Ezra, Fred Jelinek, John Lafferty, Robert Mercer, and Salim Roukos. 1992. Decision tree models applied to the labeling of text with parts-of-

speech. In *HLT '91: Proceedings of the workshop on Speech and Natural Language*, pages 117–121, Morristown, NJ, USA. Association for Computational Linguistics.

Brants, Thorsten. 2000. TnT - A Statistical Part-of-Speech Tagger. In *6th Applied Natural Language Processing (ANLP '00), April 29 - May 4*, pages 224–231. Association for Computational Lingusitics.

Brill, Eric. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Computational Linguistics*, 21(4):543–565.

Damale, M K. 1970. *Shastriya Marathi Vyaakarana*. Pune Deshmukh and Company.

Garside, Roger and Nicholas Smith. 1997. A Hybrid Grammatical Tagger: CLAWS. In Garside, Roger, Geoffrey Leech, and Tony McEnery, editors, *Corpus Annotation*, pages 102–121. Longman, London.

Giorgos, Orphanos, Kalles Dimitris, Papagelis Thanasis, and Christodoulakis Dimitris. 1999. Decision Trees and NLP: A case study in POS Tagging.

Lafferty, John, Andrew McCallum, and F. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA.

Megyesi, Beta. 1999. Improving Brill's POS Tagger For An Agglutinative Language, 02.

Oflazer, Kemal and Ilker Kuruöz. 1994. Tagging and Morphological Disambiguation of Turkish Text. In *ANLP*, pages 144–149.

Ratnaparkhi, Adwait. 1996. A Maximum Entropy Model for Part-of-Speech Tagging. In Brill, Eric and Kenneth Church, editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142. Association for Computational Linguistics, Somerset, New Jersey.

Singh, Smriti, Kuhoo Gupta, Manish Shrivastava, and Pushpak Bhattacharyya. 2006. Morphological Richness Offsets Resource Demand - Experiences in Constructing a POS Tagger for Hindi. In *Proceedings of ACL-2006*.

Tlili-Guiassa, Yamina. 2006. Hybrid Method for Tagging Arabic Text. *Journal of Computer Science 2*, 3:245–248.