Using paradigms for certain morphological phenomena in Marathi

Ashwini Vaidya
Language Technologies Research Centre
IIIT-Hyderabad
Gachibowli 500032
ashwini.vaidya@gmail.com

Dipti Misra Sharma
Language Technologies Research Centre
IIIT-Hyderabad
Gachibowli 500032
dipti@iiit.ac.in

Abstract

Words can be formed from a number of morphological operations, the commonest being inflection, derivation and compounding. Morphological analysers should be able to handle these processes, especially if they occur frequently in the language. While most morphological analysers can tackle inflectional operations easily, derivation is harder because of its less regular nature. Paradigms have been created for derived forms using XML based morhological dictionaries from the Lttoolbox package. This work builds on Akshar Bharati's (1998) paradigm-based morphological analyser for Marathi and results in an output that is richer in morphological details. The work also demonstrates how phenomena like clitics and alternations like vowel harmony can be handled using paradigms.

1 Introduction

In this paper, it will be argued that morphological operations like derivation can be taken care of using paradigms. There are some important reasons why derived forms should be analyzed dynamically rather than entered in the dictionary. Firstly, certain derivational affixes can be more productive than others and can apply to a number of forms in the language, including potential forms. Moreover, an output that is richer in morphological details is useful, especially in Machine Translation between two closely-related languages. While the word- and-paradigm method has been used successfully to take care of inflection, it can also be used to analyze the process of derivation (Blevins, 2001).

Other than derivation, the paper also discusses the use of paradigms to take care of phenomena like cliticization in Marathi as well as vowel harmony. The implementation is done using the open source tool Lttoolbox from the Apertium Machine Translation toolkit (Forcada et al., 2007). One of the reasons for choosing this tool has been its flexibility and ease with which the additional paradigms for derivation, cliticization etc. can be added to the existing morphological paradigms (The morphological analyser created by Akshar Bharati (1998) had about 91 paradigms for nouns, pronouns and numbers and 23 paradigms for verbs. The derivational paradigms were created as an additional layer on top of these paradigms).

2 Paradigms for derived forms

If we reconsider some of the characteristics of derivation, the following points are worth noting:

- Change in the category and meaning of a word after derivation
- Less regular in nature than inflection
- Operates along with inflection

The first of these characteristics, change in the category and meaning of a word is done by specifying the grammatical features in the morphological dictionary.

The second problem, that of its less regular nature is solved by listing out the root words under paradigms created specifically for derivational affixes. As this is a database driven method, listing the roots is the only solution. However, once this resource has been created, further experiments can be carried out to find criteria for attachment that are easier to model.

Finally, we have the morphological operations of inflection and derivation that operate one after the other. This is a more difficult problem to tackle

Proceedings of ICON-2009: 7th International Conference on Natural Language Processing
Macmillan Publishers, India. Also accessible from http://ltrc.iiit.ac.in/proceedings/ICON-2009

while analyzing a word as it must be segmented according to more than one affix. While tackling this aspect of derivational morphology, the Split Morphology Hypothesis (Stump, 2001) is taken into consideration. This says that inflection is usually the last operation to take place, i.e., it is never followed by derivation. According to this assumption, words could take affixes in the following order.

1. stem + inflectional suffix
2. stem + derivational suffix
3. stem + derivational suffix + inflectional suffix
4. stem + derivational suffix + derivational suffix + inflectional suffix

The sections that follow will expand upon the paradigm forms that have been created for Marathi in the morphological dictionary. The derivational paradigms are a layer built in addition to the existing inflectional paradigms in order to deal with both operations at once.

3 Using Lttoolbox

Lttoolbox (an open-source finite-state toolkit used for morphological analysis) requires the creation of a morphological dictionary that shows correspondences between surface forms (SF) and lexical forms (LF) (Garrido-Alenda and Forcada, 2002). SFs are the inflected forms of words that would be found in texts whereas LFs refer to the base forms of those words.

For instance, the word gAvAlA[1] (village +DAT) is the SF of the LF gAva + lA. This mapping al- lows the finite state transducer to process the stream of morphemes correctly, depending on whether it is analysis or generation. Lttoolbox al- lows the user to do both. The analyser will take a SF as input to return the LF and vice versa for the generator.

The regularities seen in the correspondences between SF and LF are easily encoded in the form of paradigms. The paradigms are actually rules that are organized in blocks (Stump, 2001). A group of words that belong to one paradigm will follow thesame set of spelling rules and take the same kind of affixes. A paradigm is created in the morphological dictionary file using the XML format described in the toolkit (Tyers, 2009).

Hence, within a paradigm, entry <e> encloses the correspondence between the elements <l> and <r> which stand for the left element and the right element respectively. The tag <p> includes both these left and right elements. The correspondence shows the transformation that will take place when analysis takes place. Hence, in Fig 1 SF raswyAlA will result in the analysis of raswA as the root form with the features enclosed in the <r> element.

```
<pardef n =“rasw/A_n”>
<e>
<p>
<l>A</l>
<r>A<sn = n/><sn = sg/><sn =
parsarg:0/></r>
</p>
</e>
<e>
<p>
<l>yAlA</l>
<r>A<sn = n/><sn = sg/><sn=
parsarg:lA/></r>
</p>
</e>
</pardef>
```

Figure 1. Inflectional Paradigm for Marathi raswA in the Lttoolbox format.

4 Nested Paradigms

The use of the nested paradigm is to facilitate the processes of derivation as well as derivation followed by inflection. In this way, it becomes possible to have a single dictionary entry that takes care of both derivation as well as the process of inflection that may follow it.

For instance, the nested paradigm for the adjective lahAna (small) which takes the noun forming derivational suffix -paNA is shown in Fig 2.

```
<pardef n = “lahAna/__a”>
<e>
      <p>
      <l></l>
      <r><sn =adj/></r>
      </p>

<par n =”D__/paNA”/>
</e>
</pardef>
```

Figure 2. Nested paradigm for Marathi lahAna(adj).

[1] The notation followed for Marathi is the WX format

The paradigm within lahAna calls the paradigm for paNA and hence is able to recognize a form like lahAnapaNA (smallness). On the other hand, the paradigm for a masculine noun like netA “leader” is shown in Figure 3. Here, we have the inflectional component of the noun, the subsequent derivational suffix that it calls and another layer of inflection following the derivational suffix.

Figure 2 shows an entry towards the end that has a reference to a paradigm specified in the <par> element. This can call the derivational suffix paradigm described in Figure 3.

```
<pardef n = D_/wva>
<e>
<p>
<l>wva</l>
<r>A<sn = nm/>><sn = number:eka/>
<sn = rcat:n/><sn = suff:wva/>
<sn= parsarg: 0/></</r>
</p>
</e>
<e>
<p>
<l>wvAne</l>
<r>wva<sn = nm/>><sn = number:eka/>
<sn      = rcat:n/><sn = suff:wva/>
<sn= parsarg: ne></r>
</p>
</e>
</pardef>
```

Figure 3. Paradigm for derivational suffix –tva.

In the second paradigm, newqwva or netrutva (leadership) is recognized as well as the inflection that can further attach to the derived form, which is -ne, hence netrutvAne will be analyzed.

5 Meta-paradigms

The word and paradigm approach cannot be used successfully for morphological phenomena like vowel lengthening because changes will take place away from the point of the attachment of the stem. Hence, a group of words may take the same kind of inflections but their root undergoes some change as well. For example, in Marathi, suffix -ika and -ya attaches to Sanskrit words which occur frequently in the corpus.

Hence, nisarga becomes nEsargika. The vowel becomes longer in the stem and cannot be specified in the paradigm. In order to take care of such forms, we need to make use of meta-paradigms (Forcada et al., 2007). Meta-paradigms are found in meta- dictionaries which need to be pre-processed before they can be compiled in the usual way in the dictionary compiler.

They can be created with the help of XSLT (Extensible Stylesheet Language Transformations). The stylesheets take the meta-paradigms in XML format (see Figure 4.) and output an intermediate form, which is then acted upon by another stylesheet to produce a dictionary format which can be compiled by Lttoolbox.

```
<pardef n ="p[a][viwr]/a_n">
<e>
<p>
<l><prm/><prm3/>a</l>
<r><prm/><prm3/>a<s n="adj"/></r>
</p>
</e>
<e>
<p>
<l><prm2/><prm3/>ya</l>
<r><prm/><prm3/>a<sn="n"/><sn="suff:i
ka"/>
</r>
</p>
</e>
</pardef>
```

Figure 4. Meta-paradigm for pavitra to pAvitrya.

The meta-paradigms in Figure 4 use some new elements. The <pardef/> is the same, whereas the tags <prm/> will specify the variable part in the paradigm definition. For instance, in Figure 3.4, prm, prm2 and prm3 are variable because, while changing the short e to a long E, we will need to ’move’ the portion that comes after it. Although only one vowel changes, we need to specify more than one <prm/> tag because in effect, the stylesheet generates one paradigm each for the lexical entries. The dictionary entry in this meta- dictionary (Metadix) is shown in Figure 6. From Figure 6 it is clear that ‘p’ is the only invariant part, whereas ‘a’ and ‘A’ and ‘viwra’ which specify the section that needs to be ‘moved’ will take care of the vowel lengthening.

```
<e lm="paviwra"><i>p</i>
<par n ="p[a][viwr]/a_n" prm="a"
prm2="A" prm3="viwr"/>
</e>
```

Figure 5. Dictionary entry for meta-paradigm.

After pre-processing with the XSLT stylesheets, the paradigms change to the form shown in Figure 6. Here, the original meta-paradigm has been expanded and deparametrized such that the new <pardef> contains the entire expansion of the original rule for vowel lengthening. Similar expansions will take place for any entry lemmas that belong to each of these paradigms, but with their own deparametrized <pardef> tag. The dictionary entry also changes after the application of the stylesheets here the new paradigm is now reflected in its <par> tag.

```
<pardef n="n[i][sarg]/a_n_i_E_sarg">
<e>
<p>
<l>isarga</l>
<r>isarga<s n="n"/></r>
</p>
</e>
<e>
<p>
<l>Esargika</l>
<r>isarga<s n = "adj"/><sn =
"suff:ika"/></r>
</p>
</e>
</pardef>
The dictionary entry is as follows:
<e lm="nisarga"><i>n</i>
<par n="n[i][sarg]/a_n_i_E
sarg"/></e>
```

Figure 6. Expanded paradigm for nisarga after pre-processing.

5.1 Meta-paradigms for vowel harmony

Marathi has several cases of vowel harmony where similar to the case of the addition of the deriva- tional suffix above, there are changes in the stem of the word after it undergoes inflection. For ex- ample, in Marathi, the word sUja ’swelling’ can take an inflection like -ne to get the form sujene where the long vowel is shortened. Similarly, mANUsa ’man’ becomes maNasAne. In order to handle these cases, we can use a similar solution, i,e meta-paradigms. Figure 7 shows how the meta- paradigm for a case like sUja can be created.

```
<pardef n ="s[U][j]/a_n">
<e>
<p>
<l><prm/><prm3/>a</l>
<r><prm/><prm3/>a<sn="n"/><sn="pars
arg:0"/></r>
</p>
</e>
<e>
<p>
<l><prm2/><prm3/>emaxye</l>
<r><prm/><prm3/>a<sn="n"/><sn="pars
arg:maxye"/></r>
</p>
</e>
</pardef>

Dictionary entry:
<e lm="sUja"><i>s</i>
<par n="s[U][j]/a_n" prm="U" prm2="u"
prm3="j"/>
</e>
```

Figure 7. Meta-paradigm for vowel harmony.

The stylesheets allow for more than one change in the root word i.e. it is possible to have upto 5 <prm/> tags in the meta-dictionary. In the case of vowel harmony, the number of paradigms will again increase, but the stylesheets will ensure that the work is done automatically even for a large number of suffixes.

6 Clitics

This section deals with a class of affixes which attach freely to any part of speech after the process of inflection or derivation takes place. These are not inflectional or derivational affixes but have some syntactic or discourse function. In some cases, they can be found in combination with inflectional or derivational affixes. In a language like Marathi, two clitics -hi and -ca are quite pervasive and can be found attached to nouns, verbs and adjectives. Usually, they attach to the end of the word, although there may be an exception like clitic + possessive ending + postposition in Marathi e.g. dzAdAcyAtsvara ’tree+poss+emph+above’.

In the paradigms it would be a painful process to add the clitics to each ending as it would again double the size of the paradigm. Instead, the original paradigm can be wrapped using another paradigm that contains a reference to it, while incorporating the clitic endings. Figure 8 shows this wrapping example. The clitic form is shown using a <j> tag. This shows in the output with a +'clitic' symbol to differentiate it from an ordinary affix. The <j> or 'join' tag was used by Forcada et al. (2007) to take care of phenomena like clitics as they have a function that has greater scope than the word's semantics.

```
<pardef n ="-regular_ending">
<e>
<p>
<l>Avara</l>
<r>a<sn ="n"/><sn ="ne"/><sn ="sg"/><sn
="parsarg:vara"/></r>
</p>
</e>
<e>
<p>
<l>AkhAli</l>
<r>a<sn ="n"/><sn ="ne"/><sn ="sg"/>
<sn ="parsarg:khAli"/></r>
</p>
</e>
</pardef>
<!------wrapping paradigm--------->
<pardef n="dzAd/a_n">
<e c ="no clitics">
<i/><par n="-regular_ending"/>
</e>
<!--clitic after postposition-->
<e>
<i/><par n="-regular_ending"/>
<p>
<l>ca</l>
<r><j/>ca<sn="emph_prt"/></r>
</p>
</e>
</pardef>
```

Dictionary entry:

```
<e lm ="dzAda"><i>dzAd</i><par n="dzAd/a
n"/></e>
```

Figure 8. Wrapping paradigm for clitics.

7 Resource creation for Marathi

The resource creation was carried out by extracting words from a corpus of 80,000 words. Based on the suffix or prefix that they took, the words were extracted and sorted into paradigms. These were either nested paradigms or meta-paradigms (which were created after pre-processing). The resulting morphological dictionary for Marathi had the following characteristics:

- 2000 dictionary entries for the derivation- al morph analyser
- Paradigms for 13 derivational affixes
- Wrapping paradigms for nouns and verbs to show clitic attachment

An example of the morph output was the following:

```
^ekAkIpaNA/ekAkIpaNA<n><m><sg><rcat:a
><suff:paNA>\$
^agawikapaNAne/agawikapaNA<n><m><sg><
parsarg:ne><rcat:adj><suff:paNA>\$
^OpacArika/upacAra<adj><rcat:n><suff:
ika>\$
^JAdAvaraca/JAda<n><ne><sg><parsarg:v
ara>+ca<emph>
```

Figure 9. Morph output for some forms.

8 Testing the morph

In order to evaluate the morph analyser, it is necessary to test it against all the possible derived forms in the corpus. However, this becomes harder as the size of the corpus increases. Hence, the testing process first made use of a random data set taken from the corpus to find how the morph analyser would perform on this set which would have inflectional suffixes and other particles like clitics. The second round of testing was on a smaller test set of derived forms in order to test the derived component only. The set would consist of words taken blindly from a corpus using a list of derivational suffixes.

The performance of the morph analyser was compared with the older Marathi morph analyser prepared by Bharati et al.(1998) as well as the morph analyser prepared at IIT Bombay (ILMT 2009). It was found that the morph performs better than the Akshar Bharati (1998) morph analyser but in terms of overall coverage, slightly better as compared to the IIT Bombay morph. However, the output of the morph is more detailed as compared to the IIT Bombay morph in terms of specifically identifying derivational suffixes.

The following sections describe the results obtained after testing against two data sets. The existing morphological analyser for Marathi handles derived forms to some extent, but it is done mainly through entering the word in the dictionary.

The Telugu morph analyser created at HCU, the process of derivation is handled by the Flying Lexicon method (pers. comm). The morphological analyser created in IIT Bombay also handles derivational forms to some extent (pers comm.).

8.1 Random test data

In order to evaluate the morph in a random test data environment, a corpus of 5000 words was taken for testing. It was taken from a news corpus and evaluated in comparison with the other two morph analysers (i.e Akshar Bharati morph analyser and IIT Bombay morph analyser). Another corpus, this time fom the tourism domain, consisting of about 2000 words was also used for testing. The results are shown in Table 1.

Morph	**News Corpus**	**Tourism Corpus**
Current morph	77.22 %	76.09%
Akshar Bharati morph	67.15 %	71.51 %
IIT-B Morph	75.9 %	75.29%

Table 1: Random test data results

The overall coverage of the morph shows an *average* increase of 7.35% when compared to the Akshar Bharati morph analyser across both sets. The IIT Bombay morph analyser performs about 1% less than the current morph analyser for both corpora.

However, it must be noted that not many derivational forms appear in either of the test sets. Table 2 shows the small number of derivational suffixes in this data set. Improvements in performance were because of the addition of wrapping paradigms and also pre-processing rules. Hence, the morph analyser does significantly better than the Akshar Bharati morph but only slightly better than the IIT-B morph (IIT-B morph also recognizes clitics).

The most frequently occurring derivational suffixes do not have a great presence in the test data. On the other hand, there are 244 forms in the data which consist of the clitic -ca and 112 forms consisting of the clitic -hI. Similarly, there are 198 instances of the clitic -ca and -hI in the tourism corpus. These are recognized as well because of the wrapping paradigms and are more frequent because they are more pervasive. The recognition of these forms also leads to an increase in the coverage of the current morph.

Der Affixes	**News corpus**	**Tourism corpus**
-paNa	15	0
-ika	18	1
-tA	12	5
-dAra	8	1
-paNe	11	2
Atma-	2	0
be-	3	1
-gAra	4	0
-NUka	1	0

Table 2: Affixes in the random test set

During the first round of experiments, it was observed that there were several words ending with the spoken form -aM, for example tuzhaM instead of the written form tuzhe. This problem was solved by using a pre-processing script as the aM-e alternation was regular. However, this was observed to a greater extent in the News corpus as the number of colloquial -aM ending forms were low for the tourism corpus.

The other issue was the large number of foreign i.e. English words in the news corpus. Some English words were added to the dictionary in order to increase the coverage. The problem of English words was not as acute for the tourism corpus as there were relatively fewer words. Wrapping paradigms for clitics were also implemented for nouns as well as verbs in the second round of experiments (earlier, they had been implemented only for nouns). Also, it was observed that even when the additional English words were removed from the dictionary, the number of inflected forms were numerous enough that the coverage only went down by one percent in the case of the news corpus.

Words	**% in News**	**% in Tourism**
-aM ending	7%	1%
English words	5%	1.2 %

Table 3: Presence in corpus of –aM & English words

8.2 Derived forms test data

This was an evaluation measure using a data set of approximately 250 words. These were obtained by using a news corpus of 17,000 words and checking those against the list of 13 derivational affixes that were created for the analysis. The resulting list of only derived forms was then compared with the Akshar Bharati morph analyser and the IIT Bombay morph. The current morph showed a better performance on this test set as compared to the other morph analysers. The result obtained is shown below:

Current morph analyser	61.04%
Akshar Bharati morph analyser	48.59%
IIT Bombay morph analyser	59.03%

Table 4. Derived forms test data results.

9 Error analysis

One of the main problems encountered was the presence of English words and proper nouns in the corpus. The other unknown forms were those that were not present in the dictionary that was being used. This is a shortcoming of the dictionary based method for morph analysis as the word which is not present in the dictionary is not recognized.

9.1 Random test set

In the news corpus, there were many proper nouns and place names. An almost similar number of Proper Nouns were found in the Tourism Corpus. A more effective method needs to be developed in order to deal with proper nouns in the morph analyser. Moreover, proper nouns in a language like Marathi (as well as in a number of Dravidian languages) will get inflected, resulting in an increase in the number of tokens. The proper nouns are shown in Table 5.

	News	**Tourism**
English words	3.2 %	3.8%

Table 5: English words in the corpus

The use of a named-entity recognition tool that is able to identify proper nouns would be a useful addition to the morph. The named entity recognizer that would classify the proper names into paradigms can then easily take care of the inflected proper nouns. The other unrecognized forms were because their roots were not present in the dictionary for the morph analyser.

9.2 Derived forms test set

For the derived forms data set the maximum number of unrecognized forms were for the prefix Awma-. This was mainly for the cases where prefixation and suffixation took place simultaneously. The current method of handling prefixation in Lttoolbox does not allow for inclusion of both processes, if we consider the prefix as part of the derivational analyser. Hence, Lttoolbox does not work as well for affixation that operates at both ends of a word.

The other unrecognized derivational affixes were due to the absence of these roots in the morphological dictionary and /or the absence of the derivational affix

10 Discussion

The random data set evaluation shows that although there may be derivational suffixes that are more productive in comparison to others; the occurrence of a derivational suffix in a corpus, in comparison to particles like clitics or case suffixes is lesser. This would show that productivity is different from frequency with respect to the corpus.

The current evaluation for the morphological forms relies on frequency counts, and this is especially apparent in the random test set. A more correct method of evaluating the derived component would be one that relies on productivity rather than the frequency.

Also, inflectional suffixes account for a majority of word endings in the corpus, hence in order to increase the coverage linguistically, i.e. by adding linguistic rules it becomes marginally harder. The high number of proper names, numbers and English words also would play a role in decreasing the overall presence of derived forms.

The approach followed in this paper takes the word and paradigm approach to morph analysis, where there is no morph-guessing component. However, having a named-entity recognition tool that is able to identify proper nouns would be a useful addition to the morph. The named entity recognizer that would classify the proper names into paradigms can then easily take care of the inflected proper nouns.

11 Future work

There are several ways in which future work can be carried out in this area. The first is the use of the derived forms dictionary itself. It could be used as a resource for further linguistic observations about the nature of derivation in the language. In order to find out phonological or semantic criteria for attachment of a derived suffix to a stem, this listing could be a useful tool.

There are experiments being made for learning morphological structure statistically (Goldsmith, 2001). However, in order to test and evaluate such a system, a gold standard is needed for comparison and a morph analyser prepared in this fashion would be useful for evaluation.

The power of the XML dictionaries can be adapted in order to develop a format more suitable for languages with more agglutinative morphology, where paradigms tend to become very large, in most cases with almost 2000 entries. This could be addressed by creating special suffix paradigms, similar to the ones created for prefixes, but with more flexibility to handle spelling changes.

The creation of tools to add entries to the lexicon based on knowledge of derivational affixes is also a possibility, in fact it may be possible to share a resource from a closely- related language like Hindi to extract words that have similar affixes (many Marathi and Hindi words share the affixes they have borrowed from Sanskrit and Persian).

To sum up, the knowledge of derived forms gives a more complete picture of the kind of operations that a particular word can undergo. By creating a knowledge rich morph analyser, it is possible to analyze derived forms with more morphological details. While the word and paradigm approach has some disadvantages with respect to large size of paradigms and less functionality for highly recursive morphological operations, it can nevertheless be used for handling a derivation operation followed by inflection and can also be extended to handle special cases like vowel lengthening and vowel harmony. Finally, the combination of an inflectional and derivational component would improve the performance of the morph analyser and create a useful resource for the language.

Acknowledgments

I would like to thank Francis Tyers and Jimmy O’Regan of the Apertium-stuff mailing list. Their support and prompt replies to my queries have been instrumental in writing this paper. Also, Prof Amba Kulkarni at HCU was extremely helpful and introduced me to Apertium in the first place. Any errors that remain in this paper are my own.

References

A. Bharati, A. Kulkarni, V. Chaitanya. 1998. Challenges in developing word analysers for Indian languages. Presented at the Workshop on Morphology, CIEFL, Hyderabad.

J. Blevins. 2001. Paradigmatic Derivation. *Transactions of the Philological Society*. Vol 99(2): 211-222

M. L. Forcada, B. Bonev, Ortiz S.Rojas, J.A.P Sanchez, F. G.Martinez, C. Armentano-Oller, M. Montava, and F. Tyers. 2007. Documentation of the open-source shallow-transfer machine translation platform apertium. http://xixona.dlsi.ua.es/fran/apertium2documentation.pdf.

G.T. Stump. 2001. Inflection. In eds. A. Spencer and A.M. Zwicky *The handbook of morphology*, 14–43. Blackwell.

A. Garrido-Alenda and M. L. Forcada. 2002. Comparing nondeterministic and quasideterministic finite-state transducers built from morphological dictionaries. In *Procesamiento del Lenguaje Natural (XIX Congreso de la Sociedad Espan˜ola deProcesamiento del Lenguaje Natural*, Alcal´a de Henares, Spain.

J. Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. *Computational Linguistics*, 27:2 153–198.

F. Tyers. 2009. Apertium new language pair howto. http://wiki.apertium.org/wiki/Apetrium New Language Pair HOWTO. Apertium Wikipedia article.

Indian Languages Machine Translation (ILMT) Project. 2009. http://sampark.iiit.ac.in/