DEVELOPING A RULE BASED MORPH ANALYSER FOR KASHMIRI

Aadil Amin Kak

Nazima Mehdi

Aadil Ahmad Lawaye

Department Of Linguistics

University Of Kashmir

Introduction

Morphological Analyzer is a program for analyzing the morphology of an input word, where the analyzer reads the inflected surface form of each word and provides its morphological primes.

- This morph analyser is attempted as a standalone tool with no connection with the lexicon.

- Reason for this sans-lexicon morph analyser is the unavailability (at present) of any sizable and standardized e-lexicon in Kashmiri.

Approaches

- Paradigmatic

- Rule Based Approach

- Keeping both these approaches into consideration, morph analyser for Kashmiri was developed.

PARADIGMATIC APPROACH

- The paradigm of a word (root) is the set of all of its word forms, organized by their grammatical features.

- Many words share paradigms, i.e. more than one word usually has the same inflectional behavior. So, if we identify one paradigm, and all the words share this paradigm, we will have a common analysis for all these words.

Problems Faced

- Word not found

- Some Issues with the script

RULE BASED APPROACH

- Words are analysed by using rules of the language.

- Here decision tree is used to develop rules.

Rule based approach

- The notion of rule based approach is related to the overall morphological analysis of a particular word.

⦿ Rules are built on the basis of

- Grammatical features like Gender, Number, Case, Type

  and Animacy of a word.

- Consonant-Vowel construction like CVC, CVCVC,

  CCVC etc.

Decision Tree

- Rules are implemented using Decision tree as shown in the figure given below.
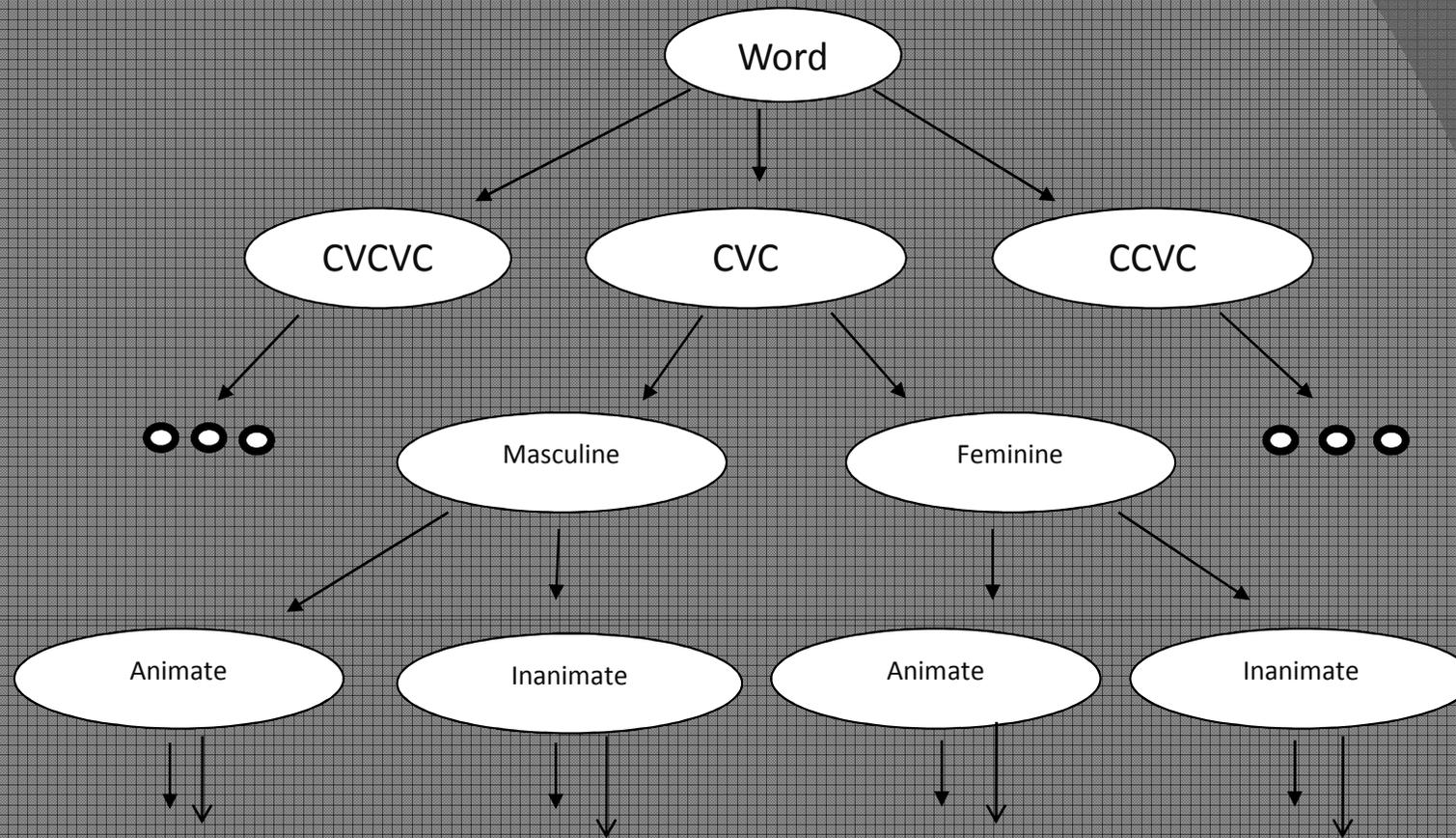
Decision Tree Implementation of Morph Analyzer.

In this scheme, words sharing the same rule i.e. having the same inflectional behavior will fall under one category. E.g. if we take the example of a CVC construction with medial vowel /a:/, words having this construction take the following inflections.

Cases	Masculine				Feminine			
	Animate		Inanimate		Animate		Inanimate	
	Singular	Plural	Singular	Plural	Singular	Plural	Singular	Plural
Nom.	∅	∅	∅	∅	∅	-I	∅	-I
Dat.	-as	-an	-as	-an	-i	-an	-i	-an
Erg.	-an	-av	-an	-av	-i	-av	-i	-av
Gen.	I-sund, I-sInz	-an-hund, -an-hInd'	-uk, -Ich	-Ik', -chi	-i-hund, -i-hInd'	-an- hund, -an- hInd'	-i-_hund, -i-hInd'	-an- hund, -an-hInd'
	-I-sInd', -I-sInzI	-an-hInz, -an-hInzI			-i-hInz, i-hInzI	-an -hInz, -an-hInzI	-i-hInz, i-hInzI	-an - hInz, -an-hInzI
Abl.	-I	-av	-I	-av	-i	-av	-i	-av

Algorithm_Generateparadigm()
Step1. Enter Root word(W), Gender(G) and Animacy(A).
Step2. Check Word construction
Step3. If the word follows CVC construction
        Function *Rule_for_CVC(W,G,A)*
      else if word follows CVCVC construction
        Fuction *Rule_for_CVCVC(W,G,A)*
       else if word follows CCVC construction
        Function *Rule_for_CCVC(W,G,A)*
Step4. Exit

Function Rule_for_CVC()
Step1. Check Vowel
        *if Vowel Is '*a:*'*
            Go to Step 2
        else

            …….
Step2. Check Gender[Masculine/Feminine]
        if Masculine
            Go to Step 3
        else if Feminine
            Go to Step 4

Step3. Check Animacy   // when root word is masculine
*   if Animate*
*   Go to Step 4*
*   else if Inanimate*
*   Go to Step 5*
Step4. The root word will take the following inflections….
*   // Masculine, Animate*

Cases	Singular	Plural
Nom.	∅	∅
Dat.	-as	-an
Erg.	-an	-av
	I_sund, I_sInz	-an_hund, -an_hInd'
Gen.		
	-I_sInd', -I_sInzI	-an_hInz, -an_hInzI
Abl.	-I	-av

Step5. The root word will take the following inflections….
       // Masculine Inanimate

Cases	Singular	Plural
Nom.	∅	∅
Dat.	-as	-an
Erg.	-an	-av
Gen.	-uk, -Ich	-Ik', -chi
Abl.	-I	-av

Step7. Check Animacy   // when root word is feminine
          if Animate
   *Go to Step 8*
          else if Inanimate
               Go to Step 9

Step8. The root word will take the following inflections….
          // Feminine Animate

Cases	Singular	Plural
Nom.	∅	-I
Dat.	-i	-an
Erg.	-i	-av
Gen.	-i-_hund, -i--_hInd' -i_ hInz, i_hInzI	-an_ hund, -an_ hInd' -an _ hInz, -an_hInzI
Abl.	-i	-av

Step9. The root word will take the following inflections….
    // Feminine Inanimate

Cases	Singular	Plural
Nom.	∅	-I
Dat.	-i	-an
Erg.	-i	-av
	-i-_hund, -i--_hInd'	-an_ hund, -an_ hInd'
Gen.		
	-i_ hInz, i_hInzI	-an _ hInz, -an_hInzI
Abl.	-i	-av

a:

→ Masculine → Animate, Inanimate

→ Feminine → Animate, Inanimate

Masculine Animate
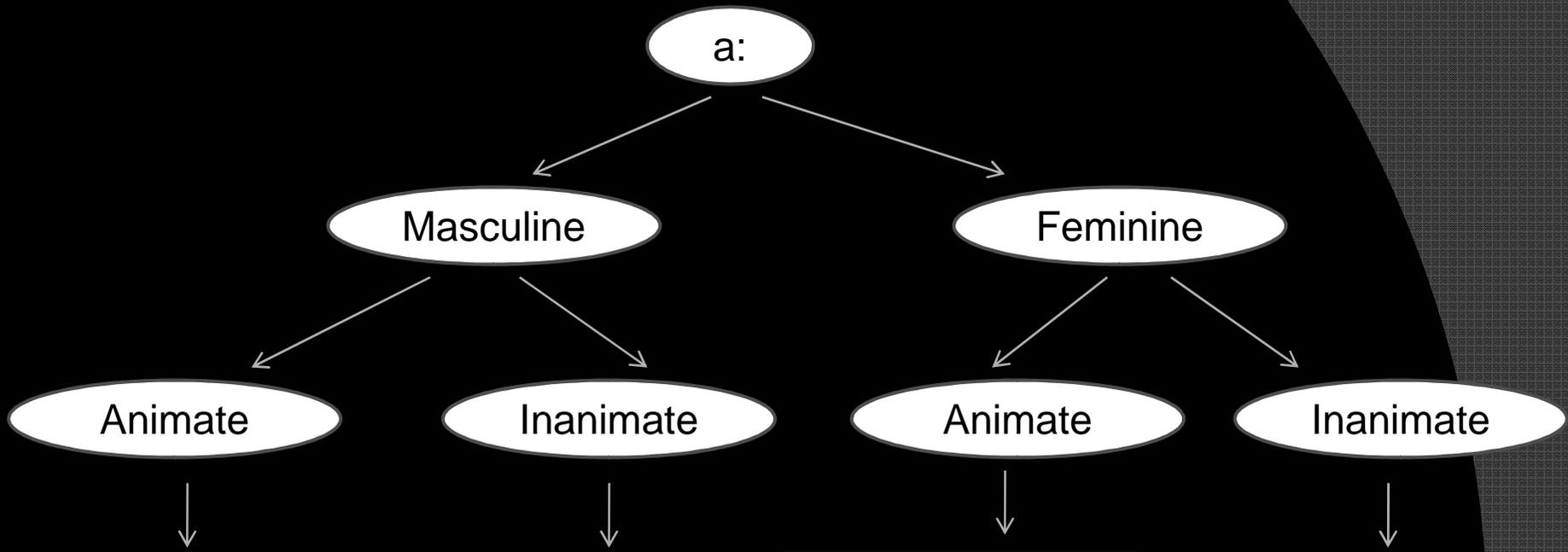
Cases	Singular	Plural
Nom.	∅	∅
Dat.	-as	-an
Erg.	-an	-av
Gen.	l_sund, l_slnz	-an_hund, -an_hlnd'
	-l_slnd', -l_slnzl	-an_hlnz, -an_hlnzl
Abl.	-l	-av

Masculine Inanimate

Singular	Plural
∅	∅
-as	-an
-an	-av
-uk, -lch	-lk', -chi
-l	-av

Feminine Animate

Singular	Plural
∅	-l
-i	-an
-i	-av
-i-_hund, -i--_hlnd'	-an_hund, -an_hlnd'
-i_hlnz, i_hlnzl	-an_hlnz, -an_hlnzl
-i	-av

Feminine Inanimate

Singular	Plural
∅	-l
-i	-an
-i	-av
-i-_hund, -i--_hlnd'	-an_hund, -an_hlnd'
-i_hlnz, i_hlnzl	-an_hlnz, -an_hlnzl
-i	-av

CONCLUSION

- Hybrid approach was used to develop a morph analyser.
- To initiate the work first paradigms were built manually and then rules were developed to overcome the issues faced while handling unknown words and to generate paradigms automatically and more efficiently.
- Work is also going on to develop a sizeable lexicon which will be given to the morph analyser to authenticate the rules.
- Till now rules for nouns are almost complete with some exceptions of course and work is also going on the analysis part of the morph analyser with the help of which the analyser will be able to analyse a word if given with its inflections.