



## RULE BASED GRAMMAR CHECKING SYSTEM FOR HINDI

LATA BOPCHE AND GAURI DHOPAVAKAR

Department of Yeshwantrao Chavan College of Engineering /Computer Technology, Nagpur, India

\*Corresponding Author: Email-

Received: January 12, 2012; Accepted: February 15, 2012

**Abstract-** This paper describes a novel method for “Hindi” grammar checking. This system utilizes a full-form lexicon for morphology analysis and rule-based systems. In this approach, we propose a system which uses a set of rules which is matched against an input Hindi sentence which has at least been POS tagged. This approach is similar to the statistics-based approach, but all the rules are developed manually in our approach.

**Keywords-** Stemmer, Morphological Analyzer Rule based Grammar Checker.

**Citation:** Lata Bopche and Gauri Dhopavakar (2012) Rule Based Grammar Checking System for Hindi. Journal of Information Systems and Communication ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, pp- 45-47.

**Copyright:** Copyright©2012 Lata Bopche and Gauri Dhopavakar. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

### Introduction

Grammar checking is one of the most widely used tools within natural language engineering applications. Most of the word processing systems available in the market incorporate spelling, grammar, and style-checking systems for English and other foreign languages. Morphological analyzer tool is used in grammar checking systems for analyzing text and POS tagging.

The morphological strength of Indian Languages warrants the use of thorough morphological analysis. It should be the first step towards any Indian language processing task. It means taking a word as input and providing the grammatical information about word. It provides information about a word's semantics and the syntactic role it plays in a sentence. It is essential for Hindi as Hindi has a rich system of inflectional morphology as like other Indo-Aryan family languages. There are basically three ways to implement a grammar checker.

**Syntax-based checking:** as described in [Jensen et al, 1993]. In this approach, a text is completely parsed, i.e. the sentences are analyzed and each sentence is assigned a tree structure. The text is considered incorrect if the parsing does not succeed.

**Statistics-based Checking:** as described in [Attwell, 1987]. In this approach, a POS-annotated corpus is used to build a list of POS

tag sequences. Some sequences will be very common (for example determiner, adjective, noun as in the old man), others will probably not occur at all (for example determiner, determiner, adjective). Sequences which occur often in the corpus can be considered correct in other texts; too, uncommon sequences might be errors.

**Rule-based checking:** as it is used in this system. In this approach, a set of rules is matched against a text which has at least been POS tagged. This approach is similar to the statistics-based approach, but all the rules are developed manually.

In Section II, we discuss previous work. In Section III explain our model and its implementation. In Section 4 concludes this paper.

### Literature Survey

Many researchers have developed System for grammar checking for Indian languages. Some of the efforts are as below:

**Bangla Grammar Checking System:** In their work related to Bangla Grammar Checking System, authors, Md. Jahangir Alam, Naushad UzZaman, and Mumit Khan explain a statistical grammar checker, which considers the n-gram based analysis of words and POS tags to decide whether the sentence is grammatically correct

or not [1]. They employed this technique for both Bangla and English and also described limitation in their approach with possible solutions.

**Punjabi Grammar Checking System:** Punjabi grammar checking system was developed by Mandeep Singh Gill and Guripreet Singh Lehal, Punjab University [2]. The grammar checking system was developed for detecting various grammatical errors in Punjabi texts. This system utilizes a full form of lexicon for morphological analysis, and applies rule-based approaches for part-of-speech tagging and phrase chunking. The system follows a novel approach of performing agreement checks at phrase and clause levels using the grammatical information exhibited by POS tags in the form of feature value pairs. The system can detect and suggest rectifications for a number of grammatical errors, resulting from the lack of agreement, order of words in various phrases etc., in literary style Punjabi texts.

**Architectural and System Design of the Nepali Grammar Checker:** This paper describes the architectural and system design of the Nepali Grammar Checker, which is in due course of research and development. The development follows a modular approach with the Grammar Checker consisting of independent modules. These modules then in turn serve as a pipeline for the over all integrated system. The Grammar Checker aims to check the grammatical errors such as nominal and verbal agreement, parts of speech inflections, phrase and clause structure and the different categories of sentence patterns for Nepali. [3]

**A rule-based Afan Oromo Grammar Checker:** Debela Tesfaye in the above titled paper describe the architectural and system design of the Afan Oromo grammar checker. Afan Oromo grammar checker has been developed and tested on real-world errors. Most of the false flags are related to compound, complex and compound complex sentences as most of the rules are constructed for simple sentences. More rules that handles the types of sentences can be added to the existing rules in order to improve the performance of the grammar checker. There are also sentences that exhibits grammatical errors but not flagged by the checker [8].

### Our Approach

For checking grammar of a Hindi sentence we have Implemented a novel algorithm.

As Hindi WordNet does not provide Hindi verb Copula and pronoun list, we have stored the entire verb copula in the verb copula lookup table, all pronouns in pronoun lookup table. Phrase Level Morphological analyzer is a part of a system.

### Phrase Level Morphological analyzer

Phrase level Morphological Analyzer provides the grammatical information about the complete sentence.

Phrase Level Morphological Analyzer architecture is divided into the following modules

#### Splitting Process

This module splits the Hindi sentence and counts the number of words present in the sentence. It stores all the words in the word [] array and counts the word in count variable.

### Pronoun Analysis

Pronoun analysis uses the PronounLookUpTable Here the entire possible pronouns are stored at the time of searching any word present in the PronounLookUpTable i.e. pronoun.

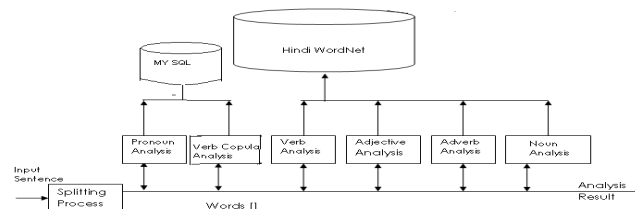


Fig. 1- Detailed Architecture of Phrase Level Morphological Analyzer

### Noun analysis

Nouns are categorized into 20 different paradigms based on

- Vowel ending.
- Valid suffix of a word.
- Gender, Number, Person and Case information.

Noun analysis uses the NounLookUpTable and Hindi WordNet. In NounLookUpTable noun is divided into two types i.e. Singular and Plural. Apply stemming process on the word then search the root word in Hindi WordNet. If the word is found in the noun list of the Hindi WordNet, it is identified as a noun. Noun analysis finds the gender, number and case information of a word.

### Verb Analysis

The Verb Group represents the following Grammatical properties:

- Aspect: Durative, Stative, Infinitive, Habitual and Perfective etc.
- Modal: Abilitive, Deontic, Probabilitative etc.
- Gender: Male, Female, Dual.
- Person: 1st, 2nd and 3rd.

Verb analysis uses the VerbLookupTable and Hindi WordNet. In VerbLookUpTable, verb is divided into two types i.e. Singular and Plural. Apply stemming process on the word and find the root word in Hindi WordNet. If word is found in the verb list of the Hindi WordNet i.e. the word is identified as a verb.

### Adjective Analysis

The adjective Group represents the following Grammatical properties:

- Aspect: Durative, Stative, Infinitive, Habitual and Perfective etc.
- Modal: Abilitive, Deontic, Probabilitative etc.
- Gender: Male, Female, Dual.
- Person: 1st, 2nd and 3rd.

Adjective analysis uses the AdjectiveLookUpTable and Hindi WordNet. In AdjectiveLookUpTable adjectives are divided into two types i.e. Singular and Plural. Apply stemming process on the word and find the root word in Hindi WordNet. If word is found in the Adjective list of the Hindi WordNet i.e. the word is identified as a Adjective.

### Adverb Analysis

The adverb Group represents the following grammatical properties:

- Aspect: Durative, Stative, Infinitive, Habitual and Perfective

- etc.
  - Modal: Abilitive, Deontic, Probabilitative etc.
  - Gender: Male, Female, Dual.
  - Person: 1st, 2nd and 3rd.
- Adverb analysis uses the AdverbLookUpTable and Hindi WordNet. In AdverbLookUpTable adverbs are divided into two types i.e. Singular and Plural. Apply stemming process on the word and find the root word in Hindi WordNet. If word is found in the Adverb list of the Hindi WordNet i.e. the word is identified as an Adverb.

**Verb Copula Analysis**

Verb copula analysis uses the VerbCopulaLookUpTable; here all the possible verb copula words are stored. At the time of searching any word in the VerbCopulaLookUpTable, identify the word as a verb copula.

**Tense analysis**

Tense analysis is performed at the phrase level Morphological analyzer. Identify the tense of an input sentence using last word present in the sentence. We have used tense analysis table shown in below.

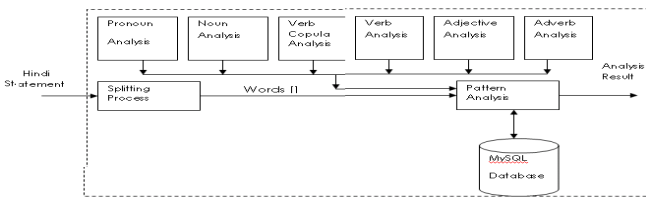
Following are the tables used in our Hindi Grammar Checker System.

- GrammaticalPatternTable
- PronounLookUpTable
- VerbCopulaLookUpTable
- NounLookUpTable
- VerbLookUpTable
- AdverbLookUpTable
- AdjectiveLookUpTable

System stores all the grammatical patterns in Grammatical Pattern Table, Pronouns in the PronounLookUpTable, Noun in NounLookUpTable, Adverb in AdverbLookUpTable, Adjective in AdjectiveLookUpTable, Verb In VerbLookUpTable.

We have used Hindi WordNet for Noun, Verb, Adjective, Adverb, and Verb. In Noun, Verb, Adjective, Adverb, Verb Lookup Table contain only singular and plural .Only these tables are used for finding singular or plural after that searching word in Hindi WordNet.

**Fig. 2-** Detail Architecture of Grammar checking system



**Proposed Algorithm**

- Step-1:** Input Hindi statement.
- Step-2:** Splitting word from Hindi sentence. (word [])
- Step-3:** Count number of words. (count)
- Step-4:** Retrieve Records from GrammaticalPatternTable (Table having Number of Pattern Element=count (Pattern))
- Step
- Step-5:** i=0
- Step-6:** if i<Pattern. Length, then Get Pattern Element from Pattern Array (PatternElement)
- | = i + 1

```

else
return null. Step 7: j=0
Step-8: if j<PatternElement (j) then p=PatternElement(j)
mainId=p.getMainId() SubId=p.GetSubId()
mainType=getMainType(mainId)
Subtype=getSubType (subId) J=j+1
Else
return Pattern[j]
Step-9: If MainType=WordMainTypeAndSubType
=WordSubType Then
Goto step 7 else
Goto step 5
Using grammar checking algorithm match the syntax of an input
sentence with available pattern. If pattern match as then it display
result with the rule id and pattern.

```

Output of Our System:  
**Result: 1:** Given statement is proper statement as per the following grammatical pattern rule.

**Rule Id: 1**

**Conclusion**

**Pattern :** <Pronoun=हैं> <Noun=SINGULAR> <Verb Copula=हैं>

Thus, as mentioned in above sections, Rule Based Hindi Grammar Checker system is successfully implemented for simple Hindi sentences. The results are very promising. The advantage of this system is the time required for grammar checking and analyzing the complete sentence is less than other existing systems. The reason of better performance is that the system only checks those patterns which have same number of words present in input sentence and does not consider all the patterns stored in database. Performance of system can be further improved by adding more rules in data base manually.

**References**

- [1] Md. Jahangir Alam, Naushad UzZaman and Mumit Khan (2006) *Ninth International Conference on Computer and Information Technology*.
- [2] Mandeep Singh Gill and Gurpreet Singh Lehal (2008) *Coling Companion volume-Poster & Demonstrations*, 149-152.
- [3] Bal Krishna Bal, Bineeta Pandey, Laxmi Khatiwada, Prajwal Rupakheti (2008) *Research Report on the Nepali grammar checker*.
- [4] Alka Choudhary, Manjeet Singh (2009) *GB Theory Based Hindi To English Translation System IEEE*.
- [5] Manish Shrivastava, Nitin Agrawal, Bibhuti Mohapatra, Smriti Singh, Pushpak Bhattacharya (2005) *The 4th Annual Inter Research Institute Student Seminar in Computer Science*.
- [6] Vishal Goyal, Gurupreet Singh Lehal (2009) *GHRCE, Nagpur*.
- [7] Bharati Akshar, Vineet Chaitanya and Rajeev Sangal (1995) *Prentice-Hall of India, New Delhi*.
- [8] Debela Tesfaye (2011) *International Journal of Advanced Computer Science and Applications*, 2(8).
- [9] Maxim Mozgovoy (2011) *Federated Conference on Computer Science and Information Systems*, 209-212.