The Hindi Named Entity Recognizer Using Hybrid Morphological Analyzer Framework

[1]Shashi Pal Singh, [2]Ajai Kumar, [3]Hemant Darbari, [4]Kanak Mohnot, [5]Neha Bansal.

[1,2,3]*AAI, CDAC, Pune, India*
[4,5]*Banasthali Vidyapith, Banasthali, India*

[1]shashipalsingh@gmail.com
[2]ajai@cdac.in
[3]darbari@cdac.in
[4]kanak.mohnot.km@gmail.com
[5]neha.bansal1391@gmail.com

***Abstract:*-Name Entity Recognition (NER) and Morphological Analyzer has been emerged as one of the Natural Language Processing (NLP) technology which is very effective and hence can be used with various kinds of applications such as Information Retrieval (IR), Question Answering (QA), Information Extraction (IE), text clustering etc. NER is basically used to identify proper nouns present in text and to classify them as different types of named entity such as people, locations, and organizations etc.**
Morphology is the field of the linguistics that studies the internal structure of the words. Morphological analysis means taking a word as input and identifying their number, gender, word formation and POS tag. Our system is used to evaluate a sentence in which we try to find the named entity in the sentence and find out the morpheme for each word in a sentence.

***Keywords*-Cleaning, Gender, HMM, Number, Named Entity, POS Tag, Rules, Tokenization Word Formation.**

I. INTRODUCTION

The large volume of text that is available on the Internet give rise to increasing interest in algorithms that can automatically process and provide required information from the text. There is a great need for a tool that extracts the exact information from the documents needed by the user. But computers cannot understand the semantics or meaning of a particular sentence or phrase,sothere is possibility of existence of many conflicts, ambiguity and portability issues. Some of the conflicts and ambiguity cases are as follows:[17]

1.When is the word "फ्रांस" being used as the name of a person and when as the name of a city?

2. "वह सुंदर लड़का है|" Is सुंदर an adjective or a name of person? [1][4]

Due to this reason a tool is developed to provide the solution of the problem.

A. Morphological Analyzer

Morphology is the study of the way words are built up from smaller meaning-bearing units, morphemes. A morpheme is often defined as the minimal meaning-bearing unit in a language. So, for example the word केला consists of a single morpheme while the केले consist of two: the morpheme केला and the morpheme –ए.Morphological Analyzer and generator is a tool for analyzing the given word and generator for generating word given the stem and its features (like affixes). [2][3][6]

B. Named Entity Recognizer

The term "Named Entity", the word Named restricts the task to those entities for which one or many rigid designators stands as referent. It is widely used in Natural Language Processing (NLP). It is the subtask of Information Extraction (IE) where structured text is extracted from unstructured text. The task of Named Entity Recognition is to categorize all proper nouns in a document into predefined classes like person, organization, location, etc. It is two step process i.e. the identification of proper nouns and its classification. Identification is concerned with marking the presence of a word/phrase as NE in the given sentences and classification is for denoting entity of the identified NE. Named entity refers to any object name in the real world. In the field of Information Extraction, Named entities mainly refers to Person, Location, Organization names and Numerical entities Time, Money, Date and Number respectively. An entity is a real world object which is distinguished from other objects. Entities are something that exists by itself, although it need not be of any material existence.[7][9][13]

II. SYSTEM DESCRIPTION

The Hindi Named Entity Recognizer Using Hybrid Morphological Analyzer Framework is developed. In this the Morphological Analyzer has four modules:- Number, Gender, Word Formation and POS tagger. For all the above modules except POS tagger, rules and corpus is used to find the information of the word. For POS tagger, hybrid approach is used. Hybrid models are basically combination of rules based and statistical models. In Hybrid system, approach uses the combination of both rule-based and ML technique and makes new methods using strongest points from each method. It is making use of essential feature from ML approaches and uses the rules to make it more efficient. [10]. The NER recognizes the entity of those words which are tagged as noun by the POS Tagger.

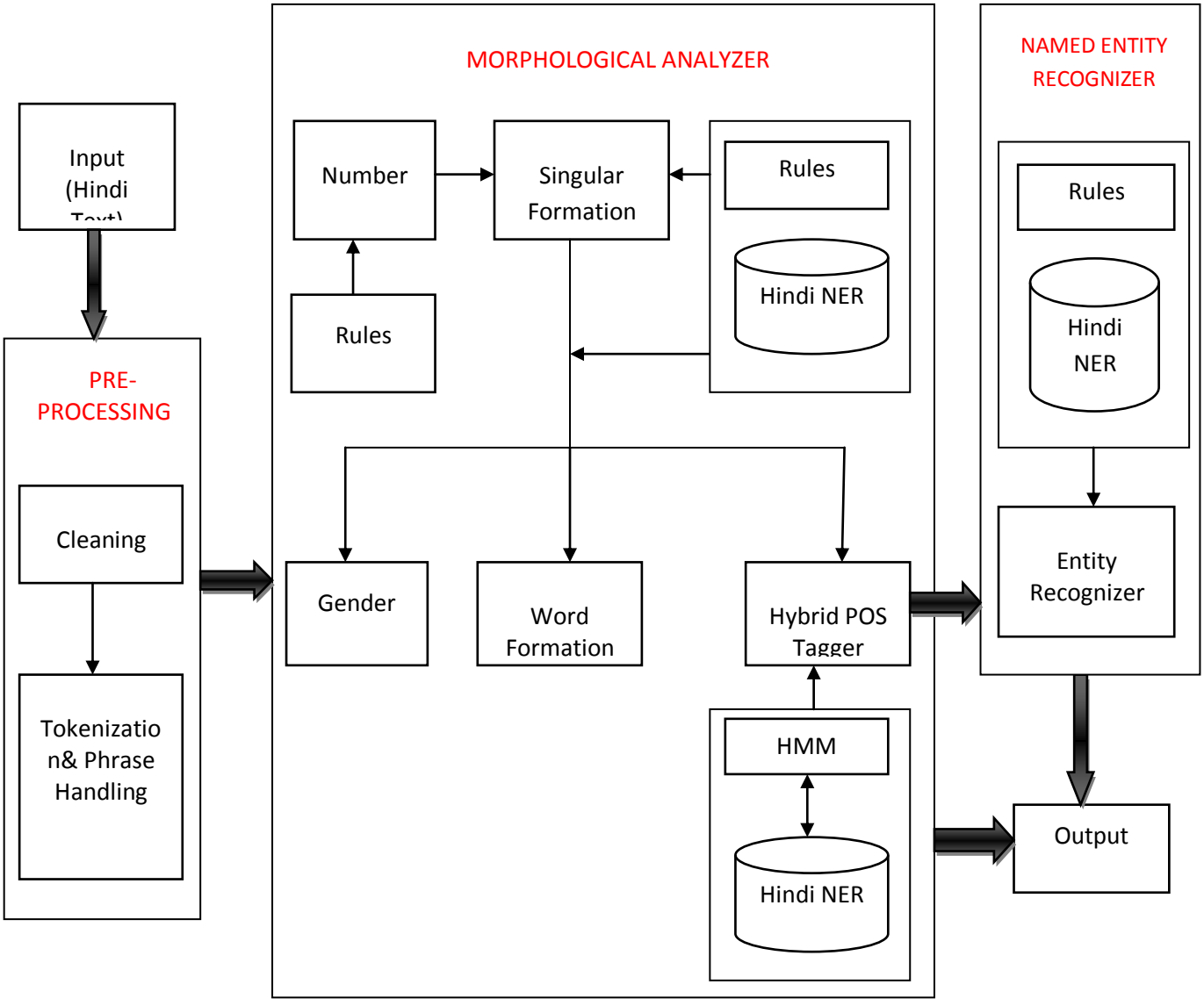

Figure 1 System Architecture

1. Input

The first step of our system is to take Hindi input text from the user. Then the text goes through various processes to produce the expected output.

2. Pre-Processing

In this block of our system the input text is pre-processed before sending it to the Morphological Analyzer. In this block we basically perform two tasks:-

2.1 Cleaning

This process is used to clean the entered Hindi input text. It removes extra spaces, full stops "|" (in Hindi called puran viram) etc. This task is accomplished to make the raw text useful.

2.2 Tokenization & Phrase Handling

In this step our system consists of breaking a stream of an input Hindi text up into meaningful elements called tokens where each token is either a word or something else like a number or a punctuation mark. In this module the system also handles phrases of name, organization, idioms etc. If two or more than two consecutive words form a phrase then we combine all these words with " " and then consider it as a single token. Therefore, we need first to keep the sentence boundaries where the sentence is something that ends with a full stop '|'(in Hindi), or a question mark"?", since the system is only able to tag entities on a token-by-token basis.

For Example:-

- भारतीयसरकार(Phrase as a single token)
- श्रीरामगुप्ता(Phrase as a single token)
- आम(Single Token)लड़का(Single Token)

3. Morphological Analyzer

This block of our system receives the input from the pre-processing block. In this block various tasks are performed which together serve as the output of the morphological analyzer. The various tasks are:-

3.1 Number

This block is used to identify the number of the input tokens. The number of the token is found by applying various types of rules.

Rule 1:-The token ending with the suffix "एँ","यों","औं" etc. are termed as plural token.

e.g. माताएँ,राजों, केले

Rule 2:-The token having a quantity before it are termed as plural token.

e.g. चारआम,एककिलोसेब

3.2 Singular Formation

This block is used to convert the plural token into the singular token. The singular is formed by applying various rules and then matching it with the database, if the match occurs then singular token is found of the respective token. The output of this block serves as the input for various others block.

Rule 1:- If the token ends with the suffix "यों" and the previous token is "इ" then we replace that with "ई" and match with the database, if the match found then singular token is found or if not then we don't replace the matra and check in database for match, if match found then the singular token found, if not then the token without removing the suffix is treated as singular.

e.g. नदियोंनदी →

3.3 Gender

In this, we identify the gender of the token i.e. male or female. The gender is assigned to the token on the basis of the rule and by checking the exception database. It's a very difficult task to assign the gender.

Rule 1: If the token ends with "ई","इया" then there is high probability of the token to be female.

e.g. नदी,रात्री,लड़की,चुहिया

Rule 2: If the token ends with "आ", "आव", "त्व" then there is high probability of the token to be male.

e.g. पिता,जमाव,महत्व

3.4 Word Formation

In this, the identification is done to find how the token is formed. With the help, of the rules and the database the formation of the token is identified.

Rule 1:- The prefixes and suffixes are identified in the token to find the token formation.

e.g. असफलताअ+सफल+ता →

संयोगसम+योग →

सुकर्मसु+कर्म →

3.5 Hybrid POS Tagger

The Hybrid POS Tagger is to assign the category to the token. To assign the category to the token various rules are formed, corpus is used and the hidden markov model is used. The various categories assigned to the token are verb, noun, adjective, adverb, post position, conjunction, particle and pronoun.

Rules Based Tagging

Rule 1:-If current token is post position then there is high probability that previous token will be noun.

e.g. उसनेपानीमेंपत्थरफेंका।

Rule 2:- If token is adjective then there is high probability that next token will be noun.

e.g. वहएकसच्चादेशभक्तहै।

Rule 3:- If word ends with तर (tar), तम (tam), इक (ik) etc. postfix then token is tagged as adjective.

For Example: - लघुतर, विशालतम, प्रामाणिक

Rule 4:- If current token is not tagged and next token tagged as an auxiliary verb, then there is high probability that current token will be main verb.

वहखानाखारहाहै।

Hidden Markov Model Tagging

We want, out of all sequences of n tags $t_1 \ldots t_n$ the single tag sequence such that $P(t_1 \ldots t_n | w_1 .. w_n)$ is highest.

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(t_1^n | w_1^n)$$

Hat ^ means "our estimate of the best one".

The Bayes rule is used to transform this equation into a set of other probabilities that are easier to compute

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$

$$\hat{t}_1^n \quad \hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(w_1^n | t_1^n) P(t_1^n)$$

Likely Hood Probability

$$P(w_1^n | t_1^n) \approx \prod_{i=1}^{n} P(w_i | t_i)$$

Prior

$$P(t_1^n) \approx \prod_{i=1}^{n} P(t_i | t_{i-1})$$

Probability

4. Named Entity Recognizer

This block is used to identify the named entity. It gets its input from the Hybrid POS tagger.

4.1 Entity Recognizer

In this module we implement NER for Hindi Language i.e. Identifies named entities like name, organization, location etc. After tagging and splitting we separate the nouns and with the help of database we compare each noun of an input string with the entity names present in the database and then finally fetched all the noun entities with their appropriate sub-entity i.e. name, place or organization. Here we have created our own database.

5. Output

This block will finally display the output from the morphological analyzer and named entity recognition block.

III. Experimental Results

For testing the morph-analyzer and the named entity recognizer an interface was developed using Java technology. This interface contains the keyboard that is provided by Google Hindi Keyboard. It is available under Google Transliteration IME that provides an input method editor which allows users to enter text in one of the supported languages using a roman keyboard.

A. Morphological Analyzer

The morphological analyzer was tested with about 500 words and 100 sentences. Some of the sample results are shown in tables 1, 2.

Input	Number	Gender	Word Formation	POS
सुंदर	एकवचन	स्त्रीलिंग	सुंदर	संज्ञा, विशेषण
केले	बहुवचन	पुलिंग	केला+ए	संज्ञा
अशालीनता	एकवचन	स्त्रीलिंग	अशालीन+ता	विशेषण
पढ़ना	एकवचन	पुलिंग	पढ़+ना	क्रिया

Table 1 Result of words tagging

Input	Number	Gender	Word Formation	POS
खाया हुआ आम राम ने खाया|	खाया हुआ-एकवचन/ आम-एकवचन/ राम-एकवचन/ ने-एकवचन/ खाया-एकवचन	खाया हुआ-पुलिंग/ आम-पुलिंग/ राम-पुलिंग / ने-पुलिंग /खाया-पुलिंग	खाया हुआ-खा+या हु+आ/ आम-आम/ राम-राम/ ने- ने /खाया-खा+या	खाया हुआ-संज्ञा / आम-संज्ञा/ राम-संज्ञा / ने-परसर्ग /खाया-क्रिया
आइएजानतेहैंदिल्लीमें हुएइसउलटफेरकीकुछव जहें	आइए-एकवचन / जानते-एकवच	आइए-पुलिंग / जानते-	आइए-आइए / जानते-जान+	आइए - क्रिया / जानते-क्रिया/हें

	न/हैं-बहुवचन / दिल्ली-एकवचन / में-एकवचन / हुए-एकवचन / इस-एकवचन / उलटफेर - एकवचन / की-एकवचन / कुछ-एकवचन / वजहें-बहुवचन	पुलिंग/ हैं-पुलिंग / दिल्ली - स्त्रीलिंग / में-स्त्रीलिंग / हुए-पुलिंग / इस-स्त्रीलिंग / उलटफेर-पुलिंग / की-स्त्रीलिंग / कुछ-स्त्रीलिंग / वजहें-स्त्रीलिंग	ते/हैं-ह+ऐ / दिल्ली-दिल +◌्ल+ई / में-में / हुए-हु+ए / इस-इस / उलटफेर-उलट+फेर / की-की/ कुछ-कुछ / वजहें-वजह+ए	- क्रिया / दिल्ली - संज्ञा / में - परसर्ग / हुए - क्रिया / इस - सर्वनाम / उलटफेर - संज्ञा / की - परसर्ग / कुछ - वेशषण / वजहें-संज्ञा
आमआदमीआमबेचता है।	आम-एकवचन/आदमी-बहुवचन / आम-एकवचन/बेचता-एकवचन/ है-एकवचन	आम-पुलिंग/ आदमी - पुलिंग/ आम-पुलिंग/ बेचता-पुलिंग/ है-पुलिंग	आम-आम/ आदमी - आदम+ई/आम-आम/बेचता-बेच+ता / है-है	आम-विशेषण/आदमी-संज्ञा/ आम-संज्ञा/बेचता-क्रिया है-क्रिया

Table 2 Result of sentence tagging

B. Named Entity Recognizer

The named entity recognizer was tested with about 500 words and 100 sentences. Some of the sample results are shown in tables 3, 4.

Input	Entity
सुंदर	व्यक्ति का नाम
केले	फल का नाम
अशालीनता	-
पढ़ना	-

Table 3 Result of word entity

Input	Entity
खाया हुआ आम राम ने खाया।	खाया हुआ-No entity/ आम-फल का नाम/ राम-व्यक्ति का नाम / ने-No entity/ खाया- No entity
आइएजानतेहैंदिल्लीमेंहुएइसउलटफेरकीकुछवजहें	आइए- No entity / जानते- No entity / हैं-No entity / दिल्ली-राज्य का नाम / में-No entity / हुए- No entity / इस-No entity / उलटफेर- No entity / की-No entity / कुछ-No entity / वजहें- No entity

आमआदमीआमबेचता है।	आम- No entity/आदमी- No entity/ आम-फल का नाम/बेचता- No entity /है- No entity

Table 4 Result of sentence entity

IV. CONCLUSION

At last we conclude that Named Entity Recognizer and Morphological Analyzer is the most important activity of

Natural Language Processing. The accuracy of any NER tool is dependent on the accuracy of morphological analyzer. Different approaches have been used by authors for the development of NER for Indian Languages. For the development of NER tool a Hindi POS tagger is required, so for this reason morphological analyzer is developed.We have shown that such a system has good performance with an ave rage accuracy of 75%-85%. We believe that further error analysis and more language specific features would improve the system performance.

REFERENCES

[1] Uma Parameshwari Rao G, Parameshwari K: CALTS, University of Hyderabad, 'On the description of morphological data for morphological analyzers and generators: A case of Telugu, Tamil and Kannada'.

[2] Harri Jappinen, 'Knowledge engineering approach to morphological analysis'**,** first conference on European chapter of the Association for Computational Linguistics.

[3] Beesley, K. and L. Karttunen. 'Finite State Morphology'. Stanford, CA: CSLI Publications, 2003.

[4] Aduriz 1., Agirre E., 'A word-grammar based morphological analyzer for agglutinative languages'**,** University of lhe Basqtlo Cotlnlry, Basque Country.

[5] Karine Megerdoomian 'Finite-State Morphological Analysis of Persia**n',** Inxight Software, Inc, University of California, San Diego.

[6] Shuly Winter, 'Hebrew Computational Linguistics: Past and Future'**, A**rtificial Intelligence Review 21**: 1**13–138, 2004, Kluwer Academic Publishers.

[7] Koskenniemi .K , Two –Level Morphology: A general Computational; Model for Word Recognition and Production, , University of Helsinki, Helsinki,1983.

[8] Lawrence R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", In Proceedings of the IEEE, 77 (2), p. 257-286February 1989.Available at: http://www.cs.ubc.ca/~murphyk/Bayes/rabiner.pdf

[9] MEMM, Shilpi Srivastava, Mukund Sanglikar & D.C Kothari. "Named Entity Recognition System for Hindi."

[10] Language: A Hybrid Approach" International Journal of Computational Linguistics (IJCL), Volume (2) : Issue (1) : 2011.Available at: http://cscjournals.org/csc/manuscript/Journals/IJCL/volume2/Issue1/IJCL-19.pdf

[11] CRF,Asif Ekbal, Rejwanul Haque, Amitava Das, Venkateswarlu Poka and Sivaji Bandyopadhyay "Language Independent Named Entity Recognition in Indian Languages" .In Proceedings of IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 33 India, January 2008.

[12] Sanjeev Kumar Sharma and Gurpreet Singh Lehal. (2011). Using Hidden Markov Model to Improve the Accuracy of Punjabi POS Tagger, Computer Science and Automation Engineering(CSAE), 2011 IEEE International Conference on June 2011, pp. 697-701.

[13] Pranjal Awasthi, Delip Rao and Balaraman Ravindran. (2006). Part Of Speech Tagging and Chunking with HMM and CRF, In the proceedings of NLPAI Contest, 2006.

[14] Dinesh Kumar and Gurpreet Singh Josan. (2010). Part of Speech Taggers for Morphologically Rich Indian Languages: A Survey, International Journal of ComputerApplications (0975 – 8887) Volume 6– No.5, September 2010, pp.1-9.

[15] Nidhi Mishra and Amit Mishra. (2011). Part of Speech Tagging for Hindi Corpus, In the proceedings of 2011 International Conference on Communication systems and Network Technologies, pp.554-558.

[16] Andrew Borthwick. 1999. "Maximum Entropy Approach to Named Entity Recognition" Ph.D. thesis, New York University.

[17] Tjong Kim Sang, Erik. F.; De Meulder, F. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proc. Conference on Natural Language Learning. pp. 142-147. Edmonton, Canada (2003).

[18] Collins, Michael and Y. Singer. 1999. "Unsupervised models for Named Entity Classification", in the proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora.

[19] F. Jelinek. 1997. Statistical Methods for Speech Recognition. MIT Press.

[20] Navneet Garg, Vishal Goyal, Suman Preet. "Rules Based Part of Speech Tagger" in the proceedings of COLING 2012:, Mumbai, December 2012. published in national & international Journals & Conference Proceedings.