Chapter 35
Review Based on Named Entity Recognition for Hindi Language Using Machine Learning Approach

Rita Shelke and Sandeep Vanjale

Abstract Named Entity Recognition (NER) is a critical job in machine learning that automatically recognizes and explains Named Things in writing, such as an Individual, a Position, or an Association. NER has played a critical role in numerous applications, including information removal and recovery, machine transformation, question answering (Q–A), and writing summarization. While much research has been conducted on NER in Hindi, no instrument with high accuracy has been created yet, as per the Literature Review. Developing a NER system for Hindi is very difficult due to the language's ambiguity, morphological richness, and resource scarcity. We provide a state-of-the-art review of different natural language processing methods (NER) for the primary language of Hindi in this article.

35.1 Introduction

The process of identifying Named Entities (NEs) from a textual document and classifying them into different conceptual categories (Name, Place, Party, Designation) is an important step in the task of Natural Language Processing (NLP). This process is called Named Entity Recognition (NER). In this age of World Wide Web, information is available in abundance but in Indian Languages, there is very less work done on NLP and Analytics per se. NER is especially a crucial proviso in applications of Information Retrieval, Machine Translation, Question Answering, Text Summarization, Efficient Search Engines, etc. [1–7]. NER development for Indian languages remains a challenge owning to syntactic and semantic complexities of Indian Languages and lack of processing tools. Since Hindi is the official language of India, it has been selected for this project to develop NER tool with Hybrid NLP Ontology and rule-based and Machine Learning-based approaches to gain better accuracy than available tools. Since Hindi is part of Indo-Aryan family, the current approach can be used

R. Shelke (✉) · S. Vanjale
Department of Computer Engineering, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, India
e-mail: rita.shelke@cumminscollege.in

S. Yadav et al. (eds.), *Proceedings of Second International Conference in Mechanical and Energy Technology*, Smart Innovation, Systems and Technologies 290,
https://doi.org/10.1007/978-981-19-0108-9_35

for other Indo-Aryan languages such as Gujarati, Bengali, Marathi, etc. as optimal NER tools are not present in the aforesaid languages as well. There are three main ways to perform NER. Using Linguistic Rule-sets along with Ontology, Machine Learning, or a Hybrid approach comprising both. Machine Learning has been the most successful in predicting unknown entities and Rule-based systems give highest accuracy [8–15].

Thus, Hybrid NER systems are most efficient for Indian languages. This system was developed to improve upon some of the issues in the current systems. Some of the issues realized are as follows:

- The accuracy of NER systems varies as per texts.
- Rule-based systems have higher accuracy but are not customizable
- Limited tag set is considered in existing systems.
- Context of word is not taken into account especially when it comes to ambiguously named entities. (Location and Person Name being same)
- Proper nouns are used as common nouns in certain cases.

In the current system, these problems are addressed by combining with indigenously built NLP algorithm using Linguistic rules and ontology depicted through RDF in tandem with the Machine Learning (ML) model of Conditional Random Fields (CRF) [16–24]. RDF has been used for creating a list of standard named entities compiled from various sources as cue to NLP algorithm and ML. 10 tags for NER have been considered whereas most of the existing systems only consider the three main tags- person, location, organization. The named entities are also phrase marked to better elicit information from the text. Bangare et al. presented cloud computing research [25–30]. Deep learning approaches were described by LMI Leo Joseph et al. [26], Athawale et al. [27] given security concerns.

35.2 Challenges

The following are the difficulties and problems that must be resolved while applying the NER tool in Hindi.

35.2.1 No Capitalization

The capitalization characteristic is critical for recognizing NEs in the English language and other foreign languages. For instance, in the phrase “Rajesh is traveling to Agra,” the English NER tool would classify “Rajesh” and “Agra” as proper nouns owing to the capitalization feature, but the Hindi language lacks this notion.

35.2.2 Morphologically Rich

The morphological analysis aims to separate a word's stems and affixes. Due to the complex morphology of Indian languages, root identification is very difficult.

35.2.3 Ambiguity

An appropriate equivocal noun is a given appellation that is also a legitimate vocabulary term that takes on its meaning when employed in the text.

35.2.4 Lack of Standardization and Spell Variations

One of the primary difficulties with Indian dialects is that various individuals in India spell similar things differently.

35.3 Previous Work and Gap Analysis

Machine Learning (ML) techniques and hybrids of Rule-Based and ML algorithms are the most often used approaches to NER. The Stanford NER tagger is an open-source NER tool that is already accessible. Stanford's classification algorithm is based on the CRF framework.

5400 Hindi words were used in the tool's analysis and testing. Accuracy and memory of 0.45 and 0.5, correspondingly, were achieved. An F-score of 0.47 was obtained. In contrast, English NER has an accuracy of 0.9223.

Chopra et al. [7] implemented NER in Hindi using the Hidden Markov Model (HMM). As previously stated, HMM aids in the development of language-independent NER systems. Scaling and analysis of these systems are equally straightforward. On 2343 training tokens and 105 testing tokens, an F-Measure of 97.14% was achieved. HMM, on the other hand, suffers from label bias. Sinha [1] makes extensive use of a small training and testing corpus.

Sinha [3] examined a total of 29 characteristics, including context words, word prefixes, word suffixes, point-of-sale information, and gazetteer lists—lists of people, places, and organizations. They get an overall accuracy of 72.78%, recall of 65.82%, and F-score of 70.45%. Sinha et al. [3] focus on applying CRF to de-ambiguate Ambiguous Proper Names (APN) in Hindi. A common noun is a name that is also an APN.

Sinha [3] also discuss the need of deriving a relevant corpus in their work. Following that, a relevant corpus was generated and classified into three categories:

names (NEP), dictionary terms not used as designations (NNE), and other confrontations (OTH). When CRF and Rule-Based outputs are merged, the resulting yield is OR-ed, a combined F-Score of 71.16% is obtained. While conducting a survey for NER in Marathi, discovered that.

Patil et al. [4] solve the issue of correctly classifying each term using a probabilistic Hidden Markov Model (HMM) trained on a manually annotated language corpus. When no pre-processing is used, the planned structure in [4] information an overall F1-Score of 62.70%, while when pre-processing is used, the proposed system reports an overall F1-Score of 77.79%. The method described in [4] correctly identified individuals, places, numbers, and units of measurement, but not other named things. The reference [4] improves efficiency via pre-processing methods such as lemmatization, which may be very expensive. The purpose of suggestion rule removal is to discover guidelines based on a collection of contacts. It assists in establishing relationships among variables in a big database.

Patawar and Potey [9] implemented NER on Marathi linguistic tweets using a hybrid method. As a widely utilized social media network, Twitter is naturally available in a variety of languages. Patawar et al. used CRF and K-Nearest Neighbor to create their method. First, we give a confidence value "cf" to the normalized tweets using a K—the value of 4. Following that, a label is assigned using the CRF labeler. The probability is calculated using CRF; however, the base label is assigned directly if "cf" is greater than. The given token is stored in the clusters and is used for further training by the system. If not, it is given a CRF label. They identify just Locations and Names, with an accuracy of 39.80 and a recall of 85.11 for location tags and 59.72 and a memory of 25.28 for designation tags. CRF simplifies the process of adding prefixes and suffixes required for NER in Indian idioms. SP has created a deep neural network-based term article recognition and classification system for the English philological. IoT and user interface work was presented by Kamal Gulati et al. [16, 25]. Bangare et al. [20] highlighted scaling challenges in machine learning.

35.4 Hindi NER Systems

The following classifications apply to the NER systems created for Hindi in the literature.

35.4.1 NER Rule-Based System

The rule-based method necessitates the creation of a set of rules by language experts. For each named entity class, a massive gazetteer list is generated using this technique. Rule-based NER arrangements are exact since they require extensive knowledge of a given linguistic and domain in order to construct syntactic-lexical pattern-based directions. Combining a rule-based and a list-based strategy, a NER arrangement for

the Hindi dialectal was produced [4]. Their algorithm discovered three new things: the money worth, the course value, and the animal/bird entity. Their organization improved previous rules by implementing a new rule called the "no-name entity rule." The list lookup method included creating several tables in the record and extracting named things from these tables. Their method has a 95.77% accuracy rating.

35.4.2 ML Based Supervised Hindi NER System

The following are the Hindi NER systems that are supervised learning-based.

35.4.2.1 Hidden Markov Model (HMM)

The authors of [5] used HMM to carry out NER in Hindi, Bengali, and Telugu. The writers used two corpora for the Hindi text: (1) At Banasthali Vidyapith, a tourist domain was established, and (2) Indian corpora of the NLTK. This method is composed of three distinct stages. The first step is annotation, in which text data is labeled. The second step is to train the HMM, which is responsible for computing the HMM's three most critical parameters: start possibility, emission possibility, and transition possibility. The third step was the HMM test stage, during which specific test phrases provided by the user were tested. Using HMM parameters, the Viterbi method calculated the optimum state sequence for the given test phrase. In [6], the authors implemented the NER system for seven distinct Indian languages using a statistical HMM-based model: Bengali, English, Hindi, Marathi, Punjabi, Tamil, and Telugu. For the Hindi language, this method achieved F-values of 0.7520. NER in Hindi was also modeled using HMM [7]. On 2343 and 105 tokens, respectively, the system was trained and tested. Their method obtained 97.14% F-measures.

35.4.2.2 Maximum Entropy Model (MaxEnt)

A NER model based on MaxEnt was created for the Hindi dialectal [8], in which the writer used a variety of characteristics, including orthographic geographies (decimal, numerals), affixes, left and right background arguments, and part-of-speech features. The authors utilized eight gazetteer lists for performance purposes, including month and weekday names, society end term lists, individual preface confrontations lists, place names lists, first names lists, middle names lists, and surnames lists. The system's presentation was compared in contradiction of a blind test set comprised of four modules: person, society, place, and date. This scheme obtained an F-Measure of 81.52%.

35.4.2.3 Support Vector Machine (SVM)

The support vector machine built a NER structure for Bengali and Hindi for two Indian dialects [9]. The system utilized many language-independent characteristics to identify named things, including background words, word affixes, expression frequency, and part-of-speech information. For Hindi, tenfold validation tests revealed recall of 80.48%, an accuracy of 74.54%, and an F-Score of 77.39%. The use of vocabulary background patterns resulted in a substantial increase of 5.13% in the F-Score. The authors of [10] created a NER structure for Hindi and the biomedical area by implementing a new kernel purpose for support vector machines. The kernel computed the weighted detachment between features based on strings.

35.4.3 ML Based Unsupervised Hindi NER System

A NER system was created to compare the co-occurrence of terms in English and Indian [18]. The authors used Wikipedia's intrinsic structure, with unstructured sheets, infoboxes, summaries, and titles, to recognize named things in Indian languages. The writers used English Wikipedia information to bootstrap the documentation of named things in Hindi and Marathi, resulting in a list of NEs. Future, this NE list was utilized to enhance the populating of multilingual entities. The NER job was completed in four steps: comparable grouping information, documentation of NEs from the infobox, documentation of NEs from the intellectual, and documentation of NEs from the text's captions. Their method was tested on a dataset of 3853 English and 2935 Hindi Wikipedia apprenticeships.

35.4.4 Deep Learning-Based Hindi NER Systems

The authors' model was created in two phases. In order to learn word embeddings using the skip-gram and glove techniques, the authors used an unlabeled dataset in the first stage. In the second stage, they employed bidirectional LSTMs. The system was qualified end-to-end utilizing labeled information after initializing the system's embedding layers with learned word vectors for each word. Their program identified Hindi properly 77.48% of the time.

35.5 Conclusion

Recognizing named entities is significant since it aids in the pre-processing stage of many natural language processing requests. This comprehensive review of Hindi NER is an excellent place for novice scholars to begin. The research demonstrates

that when a language expert is available, rule-based methods work optimally. In contrast, methods based on machine learning are more robust and straightforward to build for a language.

References

1. Mittal, N., Agarwal, B., Chouhan, G., Pareek, P., Bania, N.: Discourse based sentiment analysis for Hindi reviews. In: PReMI'13, pp. 720–725 (2013)
2. Greenwood, M., Gaizauskas, R.: Using a named entity tagger to generalise surface matching text patterns for question answering. In: Workshop on Natural Language Processing for Question Answering (2007)
3. Sinha, R.M.K.: Learning recognition of ambiguous proper names in Hindi. In: 2011 10th international conference on machine learning and applications and workshops, Honolulu, HI, pp. 178–182 (2011)
4. Patil, A.S., Patil, N.V., Pawar, B.V.: HMM based named entity recognition for inflectional language. In: Comptelix, 2017 (International Conference on Computer, Communications and Electronics 2017), pp. 565–572
5. Morwal, S., Jahan, N., Chopra, D.: Named entity recognition using Hidden Markov Model (HMM). Int. J. Nat. Lang. Comput. **1**(4), 15–23 (2013)
6. Gayen, V., Sarkar, K.: A HMM based named entity recognition system for indian Languages: The JU System at ICON 2013. In arXiv:1405.7397 (2014)
7. Chopra, D., Joshi, N., Mathur, I.: Named entity recognition in Hindi using Hidden Markov Model. In: 2nd International Conference on Computational Intelligence Communication Technology, pp. 581–586 (2016)
8. Saha, S., Sarkar, S., Mitra, P.: A hybrid feature set based maximum entropy Hindi named entity recognition. IJCNLP, 343–349 (2008)
9. Patawar, M.L., Potey, M.A.: Extending hybrid conditional random fields approachof named entity recognition for Marathi tweets. In: 2016 ICCUBEA
10. Saha, S., Narayan, S., Sarkar, S., Mitra, P.: A composite kernel for named entity recognition. Pattern Recogn. Lett. **31**(12), 1591–1597 (2010)
11. Dandapat, S., Way, A.: Improved named entity recognition using machine translation-based cross-lingual information. Int. J. Comput. Sci. Appl. **20**(3) (2016)
12. Abhinaya, N., Neethu, J., Barathi, B., Kumar, A. Soman, K.P.: AMRITA_CEN@FIRE-2014: named entity recognition for Indian languages using rich features. In: The forum for the Information Retrieval Evaluation, pp. 103–111 (2014)
13. Saharia, N.: Phone-based identification of language in code-mixed social network data. J. Stat. Manag. Syst. **20**(4), 565–574 (2017)
14. Devi, G.R., Veena, P.V., Anand Kumar, M., Soman, K.P.: Entity Extraction of Hindi-English and Tamil-English code-mixed social media text. In: Forum for Information Retrieval Evaluation. Text Processing, pp. 206–218 (2018)
15. Ekbal, A., Bandyopadhyay, S.: A conditional random field approach for named entity recognition in Bengali and Hindi. Linguistic Issues in Language Technology-LiLT **2**(1), 1–44 (2009)
16. Gulati, K., Boddu, R.S.K., Kapila, D., et al.: A review paper on wireless sensor network techniques in Internet of Things (IoT). Mater. Today Proc0 https://doi.org/10.1016/j.matpr.2021.05.067
17. Bangare, S.L., Patil, S.T., et al.: Implementing tumor detection and area calculation in MRI image of human brain using image processing techniques. Int. J. Eng. Res. Appl. **5**(4), (Part-6), 60–65 (2015). ISSN 2248-9622
18. Bangare, S.L., Patil, S.T., et al.: Reviewing Otsu's method for image thresholding. Int. J. Appl. Eng. Res. **10**(9), 21777–21783 (2015)

19. Bangare, S.L., et al.: Regenerative pixel mode and tumor locus algorithm development for brain tumor analysis: a new computational technique for precise medical imaging. Int. J. Biomed. Eng. Technol. **27**(1/2) (2018)
20. Bangare, S.L., et al.: Neuroendoscopy adapter module development for better brain tumor image visualization. Int. J. Electr. Comput. Eng. (IJECE) **7**(6), 3643–3654 (2017)
21. Kadam, R.R., Bangare, M.L.: A survey on security issues and solutions in live virtual machine migration. Int. J. Adv. Found. Res. Comput. (IJAFRC) (2012). ISSN 2348-4853
22. Chavan, S.K., Bangare, M.L.: Secure CRM cloud service using RC5 algorithm. Int. J. Comput. Trends Technol. **4**(3), 325–330 ()
23. Bangare, M.L., Joshi, S.A.: Kernel interpolation-based technique for privacy protection of pluggable data in cloud computing. Int. J. Cloud Comput. **9**(2–3), 355–374 (2020)
24. Bangare, S.L., Gupta, S., Dalal, M., Inamdar, A.: Using node. Js to build high speed and scalable backend database server. In Proceedings of NCPCI Conference, vol. 2016, p. 19 (2016)
25. Gulati, K., Sriram, V.P., Sharma, M., Parul, Eliyas, S., Bangare, S.L.: Use for graphical user tools in data analytics and machine learning application. Turk. J. Physiotherapy Rehab. **32**(3). ISSN 2651-4451, e-ISSN 2651-446X
26. Leo Joseph, L.M.I., Bangare, S.L., et al.: Methods to identify facial detection in deep learning through the use of real-time training datasets management. EFFLATOUNIA—Multidiscip. J. **5**(2), 1298–1311. ISSN 1110-8703
27. Athawale, S., Giri, V., Bangare, S.L.: Collateral extension in provocation of security in IoT. Int. J. Future Gener. Commun. Netw. **14**(1), 3703–3716. ISSN 2233-7857, Web of Science
28. Bangare, S.L.: Brain tumor detection using machine learning approach. Design Engineering (7), pp. 7557–7566. ISSN 0011-9342, Scopus Index- Q4
29. Bangare, M.L., Bangare, P.M., Apare, R.S., Bangare, S.L.: fog computing based security of IoT application. Design Engineering (7), pp. 7542–7549. ISSN 0011-9342, Scopus Index-Q4
30. Bangare, S.L., Bangare, P.S., Patil, K.P.: Internet of things with green computing. Turk. J. Physiotherapy Rehab. **32**(3), 12494–12497. ISSN 2651-4451, e-ISSN 2651-446X