IIT(BHU)@IECSIL-FIRE-2018: Language Independent Automatic Framework for Entity Extraction in Indian Languages

Akanksha Mishra, Rajesh Kumar Mundotiya, and Sukomal Pal

Indian Institute of Technology, Varanasi, India
{akanksham.rs.cse17, rajeshkm.rs.cse16, spal.cse}@itbhu.ac.in
http://www.iitbhu.ac.in/

Abstract. The paper discusses about our work submitted to the track IECSIL [3] organized by ARNEKT in conjunction with Forum for Information Retrieval and Evaluation 2018. The track primarily focuses on developing language independent system for information extraction on Indian languages. We focused on the identification and categorization of entities in the text. We used word embedding for the feature representation. We proposed Bidirectional LSTM recurrent neural network model for entity extraction from the provided text for the five Indian languages such as Hindi, Kannada, Malayalam, Tamil and Telugu. The proposed technique is evaluated in terms of two metrics, accuracy and F1-score.

Keywords: Named Entity Recognition, Recurrent Neural Network, Word Embedding

1 Introduction

With the growing information and huge availability of structured and unstructured data, it is important to extract relevant information from the available text. The relevant information is further identified and useful keywords are extracted within the digital texts. The term which defines the actual meaning of the sentence and acts as important nouns in the sentence are extracted for information. These nouns are further categorized to determine datenum, event, location, name, number, organization, occupation, things and other information.
In the subsequent sections, we discuss objective of the task, statistics about corpus, present an overview of the framework, and analyze performance of the developed system.

2 Related Work

We conducted literature review for one of the fundamental task in the domain of Natural Language Processing. Named Entity Recognition can be performed broadly using three approaches namely rule-based, machine learning and deep learning based approaches. However, there is requirement of language expertise

to develop rule-based techniques for entity extraction.
Asif et al. [5] proposed statistical conditional random fields (CRFs) based approach for some of the Indian Languages. The system used both language dependent and independent features. Also, linguistic features for some of the languages were extracted from gazetteers list. Another work done by Asif et al. [4] used support vector machine based approach for the named entity recognition for Hindi and Bengali Languages. Also, lexical context patterns were generated using unsupervised algorithms. Sujan et al. [9] developed named entity recognition system for Hindi language using hybrid set of features for Maximum Entropy (Max-Ent) method.
A survey about Named Entity Recognition systems for Indian and Non-Indian Languages is done by Nita et al. [8]. They summarized different rule-based and statistical techniques used for the Indian Languages. Deep learning based approaches are explored for English and some other languages for named entity recognition however not much work is done generic to Indian Languages. Vinayak et al. [1] proposed recurrent neural network based approach for named entity recognition in English and Hindi language.

3 Task Description and Corpus

There are two tasks scheduled in ARNEKT-IECSIL track for information extraction. First task is to identify and categorize entity within the conversational systems. Second task is to determine relation between the entities extracted by the first task. We built the system for entity extraction only.
The corpus comprises five Indian languages (Hindi, Kannada, Malayalam, Tamil and Telugu) and is provided by the task organizers [2]. Corpus is divided into Train (60%), Test-1 (20%) and Test-2 (20%) set. Table 1 lists statistics related to Train, Test-1 and Test-2 set for Named Entity Recognition. The corpus is built in such a way that it will support systems built independent of languages.

Table 1: Corpus Statistics for Entity Extraction

Indian Languages	**#lines in Train data**	**#lines in Test-1 data**	**#lines in Test-2 data**
Hindi	1,548,570	519,115	517,876
Kannada	318,356	107,325	107,010
Malayalam	903,521	301,860	302,232
Tamil	1,626,260	542,225	544,183
Telugu	840,908	280,533	279,443

4 Methodology

This section discusses about the implementation of our approach. The architecture of our model is shown in Figure 1. In our proposed technique, we generated

vector representation of words and tags which is fed to the bidirectional Long Short-Term Memory recurrent neural network to train the model and predict tags for test data. This section further describes about the feature representation and model description.

4.1 Feature Representation

Word embedding is developed to capture information about the words in the text corpus using Word2Vec [6, 7]. We used continuous bag of words training algorithm to learn the word embedding. We considered all the words present in the corpus to generate word embedding. Each sentence is tokenized and further each token is represented into 100-dimensions to construct the embedding.
Different categories of entity is represented as binary vectors using One Hot Encoding. This requires all classes to be represented as integers which is further represented as binary vector where presence of integer is marked as 1 otherwise 0.

4.2 Model Description

Word representation discussed earlier is used as input to BiLSTM layer. Our proposed model uses BiLSTM to learn the contextual relationship between words from past and future context. Word representation obtained during feature representation stage is given as input to the model. BiLSTM layer uses 2 LSTM layers having number of units as 100 in each layer. One of the LSTM layer is connected in the forward direction which takes input in the same sequence while the other is connected in the backward direction which takes reverse copy of the input sequence. Output obtained by both the layers is concatenated to produce embedding for the input token. Further, we explore various activation functions and varying recurrent dropout at BiLSTM layer.
Dropout [10] with varying rate is applied between the BiLSTM layer and output layer on the hidden neurons. The output obtained is fed to dense layer having 10 units for predicting classes of the entity. Our approach follows a series of steps as given below in Algorithm 1.

5 Experiments

We used Keras[1] neural network library to implement bidirectional Long Short-Term Memory recurrent neural network which uses either Tensorflow[2] or Theano[3] as backend. The model is trained for 10 epochs with batch size of 32. Hyperparameters used in the model is shown in Table 2.

[1] https://keras.io/
[2] https://www.tensorflow.org/
[3] http://deeplearning.net/software/theano/

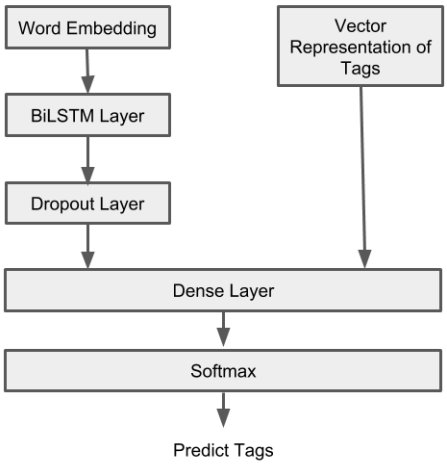

Fig. 1: Final Architecture of the System

Algorithm 1: Framework for Entity Extraction

Data: Train and Test data
Result: Predict type of entity for tokens in Test data
1 Read Train dataset and Test dataset
2 Generate nested list of words and tags of each sentence in Train dataset
3 Generate nested list of words of each sentence in Test dataset
4 Determine maximum length of sentence in Train and Test dataset
5 Generate list of unique tags and unique words
6 Represent unique tags using one hot encoding
7 Develop word embedding using nested list of words of Train dataset
8 Create dictionary with key as word if it exists and value as vector representation of that word
9 Perform padding on Train and Test dataset based on maximum length obtained in Step 4
10 Train model comprises BiLSTM, Dropout and Dense layer
11 Predict tags for Test tokens

Table 2: Hyper-parameters for the model

Hyper-parameters	**Values**
Word Embedding Dimension	100
Recurrent dropout	0.2
BiLSTM layer activation	tanh
Dropout layer	0.3
Dense layer activation	softmax
Optimizer	adam
Loss	categorical crossentropy

6 Results

The evaluation of the proposed technique is carried on the Test-1 and Test-2 corpora provided by the task organizers. The evaluation is divided into two stages to help the participants to test the system built in real time. Pre-evaluation was performed on Test-1 corpora and final-evaluation was performed on Test-2 corpora. Ranking is predicted based on the accuracy measure by taking average of accuracy obtained for all Indian languages.
Baseline system built on Naive Bayes Classifier was released by the task organizers during pre-evaluation stage. We achieved 5.45% better accuracy as compared to the baseline system for Test-1 corpora. Accuracy obtained during the pre-evaluation stage for different languages is shown in Table 3. Accuracy is calculated by comparing the result obtained by the proposed system with the labelled data provided by the organizers.
For analyzing the performance of the system on Test-2 corpora, task organizers evaluated the submissions on two metrics, accuracy and F1-score. We submitted three submissions for final-evaluation on Test-2 corpora. Detailed statistics about the accuracy obtained in different submissions for different Indian languages is listed in Table 4. It is observed that Submission 3 performs better than other two submissions. Also, F1-score is calculated by the task organizers for Test-2 corpora. Performances of our 3 submissions are shown graphically for F1 metric in Figure 2, 3 and 4.

Table 3: Accuracy for Test-1 corpora

Indian Languages	**Submission**
Hindi	94.40
Kannada	90.09
Malayalam	89.97
Tamil	91.23
Telugu	90.20
Average	91.18

Table 4: Accuracy for Test-2 corpora

Indian Languages	**Submission 1**	**Submission 2**	**Submission 3**
Hindi	94.45	94.47	94.92
Kannada	89.53	89.85	89.88
Malayalam	89.04	88.89	89.13
Tamil	90.46	90.40	90.47
Telugu	90.04	90.04	90.32
Average	90.70	90.73	90.94

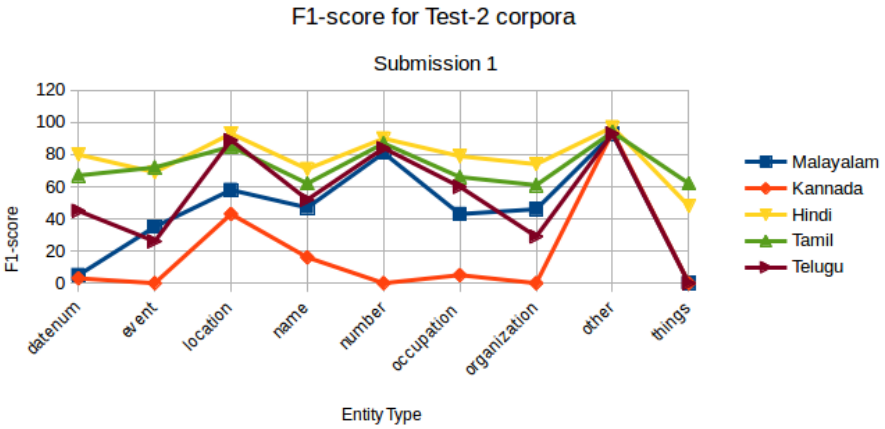

Fig. 2: F1-score for Test-2 corpora for Submission 1

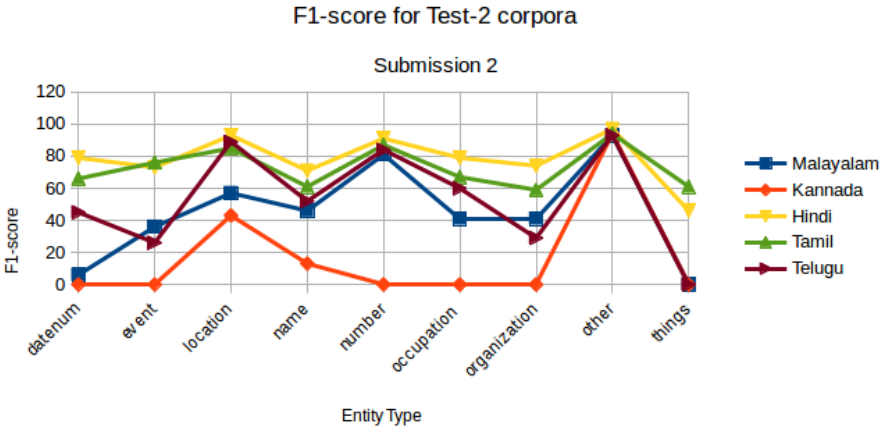

Fig. 3: F1-score for Test-2 corpora for Submission 2

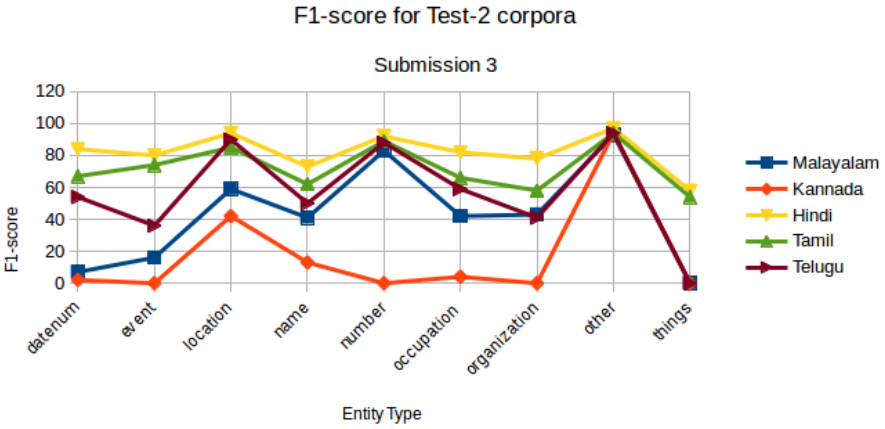

Fig. 4: F1-score for Test-2 corpora for Submission 3

7 Conclusion and Future Work

We proposed a fully automatic system for entity extraction in Indian Languages namely Hindi, Kannada, Malayalam, Tamil and Telugu. We obtained 91.18% accuracy during pre-evaluation stage on Test-1 corpora and 90.94% accuracy during final-evaluation stage on Test-2 corpora. The system can be improved either by incorporating language specific features or probably by representing tokens at character level.

8 Acknowledgements

We would like to thank the organizers for giving us an opportunity to work on the challenging task, providing us the guidelines and support.

References

1. Athavale, V., Bharadwaj, S., Pamecha, M., Prabhu, A., Shrivastava, M.: Towards deep learning in hindi NER: an approach to tackle the labelled data sparsity. CoRR **abs/1610.09756** (2016), `http://arxiv.org/abs/1610.09756`
2. Barathi Ganesh, H.B., Soman, K.P., Reshma, U., Mandar, K., Prachi, M., Gouri, K., Anitha, K., Anand Kumar, M.: Information extraction for conversational systems in indian languages - arnekt iecsil. In: Forum for Information Retrieval Evaluation (2018)
3. Barathi Ganesh, H.B., Soman, K.P., Reshma, U., Mandar, K., Prachi, M., Gouri, K., Anitha, K., Anand Kumar, M.: Overview of arnekt iecsil at fire-2018 track on information extraction for conversational systems in indian languages. In: FIRE (Working Notes) (2018)

4. Ekbal, A., Bandyopadhyay, S.: Named entity recognition using support vector machine: A language independent approach. International Journal of Electrical, Computer, and Systems Engineering **4**(2), 155–170 (2010)
5. Ekbal, A., Haque, R., Das, A., Poka, V., Bandyopadhyay, S.: Language independent named entity recognition in indian languages. In: Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages (2008)
6. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
7. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
8. Patil, N., Patil, A.S., Pawar, B.: Survey of named entity recognition systems with respect to indian and foreign languages. International Journal of Computer Applications **134**(16) (2016)
9. Saha, S.K., Sarkar, S., Mitra, P.: A hybrid feature set based maximum entropy hindi named entity recognition. In: Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I (2008)
10. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research **15**(1), 1929–1958 (2014)