# Bengali Named Entity Recognition using Margin Infused Relaxed Algorithm

Somnath Banerjee, Sudip Kumar Naskar, and Sivaji Bandyopadhyay

Department of Computer Science and Engineering,
Jadavpur University, India
s.banerjee1980@gmail.com
{sudip.naskar,sbandyopadhyay}@cse.jdvu.ac.in

**Abstract.** The present work describes the automatic recognition of named entities based on language independent and dependent features. Margin Infused Relaxed Algorithm is applied for the first time in order to learn named entities for Bengali language. We used openly available annotated corpora with twelve different tagset defined in IJCNLP-08 NERSSEAL shared task and obtained 91.23%, 87.29% and 89.69% precision, recall and F-measure respectively. The proposed work outperforms the existing models with satisfactory margin.

## 1 Introduction

Named entities (NEs) have a special status in Natural Language Processing (NLP) because of their distinctive nature which other elements of human languages do not have, e.g. NEs refer to specific things or concepts in the world and are not listed in the grammars or the lexicons. Automatic identification and classification of NEs benefits in text processing due to their significant presence in the text documents. Named Entity Recognition (NER) is a task that seeks to locate and classify NEs in a text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, etc. The NER task can be viewed as a two stage process: a) Identification of entity boundaries, b) Classification into the correct category. For example, if *"Sachin Tendulkar"* is a named entity in the corpus, it is essential to identify the beginning and the end of this entity in the sentence. Following this step, the entity must be classified into the predefined category, which is PERSON (Named Entity Person) in this case.

The NER task has important significance in many NLP applications such as Machine Translation, Question-Answering, Automatic Summarization, Information Extraction etc. The task of building an NER for Indian languages (ILs) presents various challenges related to their linguistic characteristics. Some of them are: no capitalization, unavailability of large gazetteer, relatively free word order, spelling variation, rich inflection, ambiguity, etc. In this work, we identified suitable language independent and dependent features for the Bengali NER task and used *Margin Infused Relaxed Algorithm* (MIRA) to develop NER system for Bengali.

P. Sojka et al. (Eds.): TSD 2014, LNAI 8655, pp. 125–132, 2014.
This is preprint prepared by Proceedings editor for Springer International Publishing Switzerland.

## 2 Related Work

Bandyopadhyay[1] stated that the computational research aiming at automatically identifying NEs in texts forms a vast and heterogeneous pool of strategies, techniques and representations from hand-crafted rules towards machine learning approaches. Most of the previous NER systems are based on one of the following approaches:

– Linguistic approaches

– Machine Learning(ML) based approaches

– Hybrid approaches

The linguistic approaches based NER systems ([2, 3, 4]) typically use hand-crafted grammatical rules written by linguistics. On the other hand, ML based NER systems use learning algorithms that require large annotated datasets for training and testing [5]. ML methods such as Hidden Markov Model (HMM) [6], Conditional Random Field (CRF) [7], Support Vector Machine (SVM) [8], Maximum Entropy (ME) [9] are the most widely used approaches. Besides the above two approaches, Hybrid approaches based NER systems [10] combines the strongest point from both the Rule based and statistical methods.

Mainly, ML and hybrid approaches were used successfully in NER for Bengali language. The survey by [11] on NER for ILs detailed the various approaches used for Bengali NER by researchers.

*ML-based*: [12],[13],[14],[15],[16],[17],[18].

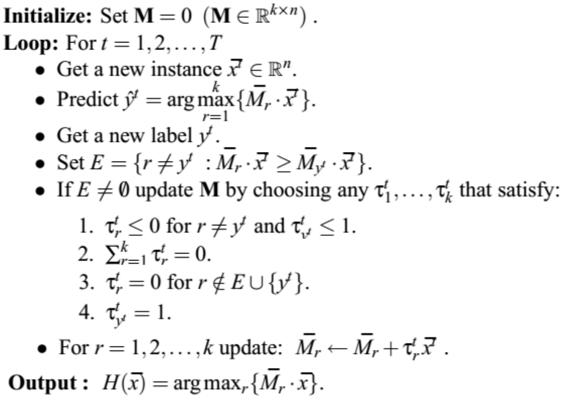
*Hybrid-based*: [10],[19],[20].

Though a few use of MIRA was noted for English [21], but it has not been used in NER for any Indian languages till date. This is one of the reasons to use MIRA in this work.

## 3 Margin Infused Relaxed Algorithm

Crammer and Singer [22] reported *Margin Infused Relaxed Algorithm* is a machine learning algorithm for multiclass classification problems. It is designed to learn a set of parameters (vector or matrix) by processing all the given training examples one-by-one and updating the parameters according to each training example, so that the current training example is classified correctly with a margin against incorrect classifications at least as large as their loss. The change of the parameters is kept as small as possible. MIRA is also called passive-aggressive algorithm (PA-I), is an extension of the perceptron algorithm for online machine learning that ensures that each update of the model parameters yields at least a margin of one. The flow of the MIRA is depicted in Figure 1.

Suppose sequence $(\bar{x}^1, y^1), ..., (\bar{x}^t, y^t), ...$ is the instance-label pairs. Each instance $\bar{x}^t$ is in $\mathbb{R}^n$ and each label belongs to a finite set $Y$ of size $k$. It can be assumed without loss of generality that $\mathtt{Y} = \{1, 2, ..., k\}$. A multiclass classifier is a function $H(\bar{x})$ that maps instances from $\mathbb{R}^n$ into one of the possible labels in $\mathtt{Y}$. The classifier is in the form $H(\bar{x})$ = argmax$_{r=1}^{k}\{\bar{M}_r.\bar{x}\}$, where **M** is a $k \times n$ matrix over the reals and¯$\bar{M}_r \in \mathbb{R}^n$ denotes the r$^t h$ row of **M**. The inner product of $\bar{M}_r$ with the instance $\bar{x}$ is called the



```
**Initialize:** Set $\mathbf{M} = 0$ $(\mathbf{M} \in \mathbb{R}^{k \times n})$ .
**Loop:** For $t = 1, 2, \ldots, T$
    • Get a new instance $\vec{x^t} \in \mathbb{R}^n$.
    • Predict $\hat{y}^t = \arg\max_{r=1}^{k} \{\bar{M}_r \cdot \vec{x^t}\}$.
    • Get a new label $y^t$.
    • Set $E = \{r \neq y^t \ : \bar{M}_r \cdot \vec{x^t} \geq \bar{M}_{y^t} \cdot \vec{x^t}\}$.
    • If $E \neq \emptyset$ update $\mathbf{M}$ by choosing any $\tau_1^t, \ldots, \tau_k^t$ that satisfy:
        1. $\tau_r^t \leq 0$ for $r \neq y^t$ and $\tau_{y^t}^t \leq 1$.
        2. $\sum_{r=1}^{k} \tau_r^t = 0$.
        3. $\tau_r^t = 0$ for $r \notin E \cup \{y^t\}$.
        4. $\tau_{y^t}^t = 1$.
    • For $r = 1, 2, \ldots, k$ update: $\bar{M}_r \leftarrow \bar{M}_r + \tau_r^t \vec{x^t}$ .
**Output :** $H(\vec{x}) = \arg\max_r \{\bar{M}_r \cdot \vec{x}\}$.
```

**Fig. 1.** Algorithm (Crammer and Singer [22])

*similarity-score* for class $r$. Thus, the considered classifiers set the label of an instance to be the index of the row of **M** which achieves the highest *similarity-score*.

On round $t$ the learning algorithm gets an instance $\bar{x}^t$. Given $\bar{x}^t$, the learning algorithm outputs a prediction, $\hat{y} = \arg\max_{r=1}^{k}\{\bar{M}_r.\bar{x}^t\}$. It then receives the correct label $y^t$ and updates its classification rule by modifying the matrix **M**. It can be said that the algorithm made a (multiclass) prediction error if $\hat{y}^t \neq \bar{y}^t$. The goal is to make as few prediction errors as possible.

We used *miralium*[1] which is the open source java implementation of MIRA.

## 4 Features

The success of any machine learning algorithm depends on finding an appropriate combination of features. This section outlines language dependent and language independent features.

### 4.1 Language Independent Features

Language independent features can be applied to any language including ILs, e.g., Bengali, Hindi, Tamil, Punjabi, etc. The following language independent features are applied to this work.

**Window of words**: Preceding or following words of the target word might be used to determine its category. The previous m words and next n words along with target word are considered to build the window. But, it has been observed that majority of research works used $m = n$. Following a few trials we found that a suitable window size is five with $m = 2$ and $n = 2$.

**Word Suffix**: Target word suffix information is very helpful to identify NEs especially for highly inflectional languages like ILs. Though the stemmer or morphological

[1] `https://code.google.com/p/miralium/`

analyzer recognizes the suffix properly, but in the absence of those fixed length suffix can be used as a feature. We used four fixed length suffixes of length 5, 4, 3 and 2 respectively.

**Word prefix**: Target word prefix also can be used like prefix feature. We used four fixed length prefixes of length 5, 4, 3 and 2 respectively.

**First word**: First word of a sentence can be used as a feature because in most of the languages the first word is the subject of the sentence.

**Word length**: It has been observed that short words are rarely NEs. So, length of the word may be used as a feature.

**Part of Speech (POS)**: The POS of the target word and surrounding words may be useful feature for NER. Since NEs are noun phrases, the noun tag is very relevant.

**Presence of Digit**: Presence of digit in the target word is a very useful feature. This feature is very helpful to identify time expression, measurement and numerical quantities. Most of the cases, digit combines with symbols make NEs, e.g., 12/10/2014, 55.44%, 22/-, etc.

### 4.2 Language Dependent Features

Language dependent features are increasing the accuracy of the NER systems. So, in most of the experiments they are used along with language independent features.

**Clue words**: Clue words play a useful role to determine NEs. They occur before or after the NEs. For example, *'Mr.'* is most likely present before starting a person name. Similarly, *'Limited'* is most likely present after an organization. List of clue words can be prepared for NER. In this work, we prepared two clue word lists, namely person clue list and organization clue list under human supervision from the archive (100 documents) of an online available widely used Bengali newspaper. Person and organization clue word lists contain 39 and 53 words. This feature is used as binary feature. If the target word present in the lists, then the value is set to 1, otherwise 0.

**Gazetteers list**: Lists of names of various types are helpful for NER. We manually prepared four lists, namely names of months, names of sessions, Days of a week, names of units.

## 5 Experiments

This section describes our study of NER using language independent and dependent features applying MIRA and comparative study with the existing Bengali NER models. We considered the same baseline system (i.e., name finder tool) reported in [23] which is an open source, maximum entropy based and part of OpenNLP[2] package. At first, experiments were performed with language independent features only. Then we used language independent and dependent features together. It has been observed that the use of language dependent features increase the overall F-measure (3.12%).

[2] `http://opennlp.sourceforge.net/`

### 5.1 Corpus and Tagset

We used IJCNLP-08 NER on South and South East Asian Languages (NERSSEAL) *shared task data*[3]. The shared task data is tagged with the Tagset[4] which consists of 12 NE tags. Corpus statistics, tagset and tagset statistics are shown in Table-1 and Table-2 respectively.

**Table 1.** Corpus Statistics.

| | **Training** | **Testing** |
|---|---|---|
| Sentences | 6030 | 1835 |
| Words | 112845 | 38708 |
| NEs | 5000 | 1723 |

**Table 2.** NERSSEAL NE Tagset and Statistics.

| **Tag** | **Name** | **Example** | **Training** | **Testing** |
|---|---|---|---|---|
| NEP | Person | Bob Dylan | 1299 | 728 |
| NED | Designation | President, Chairman | 185 | 11 |
| NEO | Organization | State Government | 264 | 20 |
| NEA | Abbreviation | NLP, I.B.M. | 111 | 9 |
| NEB | Brand | Pepsi, Windows | 22 | 0 |
| NETP | Title-Person | Mahatma, Dr., Mr. | 68 | 57 |
| NETO | Title-Object | American Beauty | 204 | 46 |
| NEL | Location | New Delhi, Paris | 634 | 202 |
| NETI | Time | 10th July, 5 pm | 285 | 46 |
| NEN | Number | 3.14, Fifty five | 407 | 144 |
| NEM | Measure | three days , 5 kg | 352 | 146 |
| NETE | Terms | Horticulture | 1165 | 314 |

### 5.2 Results

The performance of the system is evaluated in terms of the standard precision, recall, and F-Measure as follows:

Precision: $P = \frac{c}{r}$
Recall: $R = \frac{c}{t}$
F-Measure: $F_{\beta=1} = \frac{2 \times P \times R}{P+R}$

where $c$ is the number of correctly retrieved (identified) NEs, $r$ is the total number of

[3] http://ltrc.iiit.ac.in/ner-ssea-08/index.cgi?topic=5

[4] http://ltrc.iiit.ac.in/ner-ssea-08/index.cgi?topic=3

**Table 3.** Evaluation for specific NE tags. (NP: Not present in reference data)

| | Lang. Independent | | | Lang. Dependent | | |
|---|---|---|---|---|---|---|
| **Tag** | P | R | $F_{\beta=1}$ | P | R | $F_{\beta=1}$ |
| NEP | 92.86 | 89.29 | 91.04 | 94.20 | 91.48 | 92.82 |
| NED | 66.67 | 36.36 | 47.06 | 70.00 | 63.64 | 66.67 |
| NEO | 73.68 | 70.00 | 71.79 | 78.95 | 75.00 | 76.92 |
| NEA | 37.50 | 33.33 | 35.29 | 42.86 | 33.33 | 37.50 |
| NEB | NP | NP | NP | NP | NP | NP |
| NETP | 82.35 | 73.68 | 77.78 | 83.93 | 82.46 | 83.19 |
| NETO | 80.95 | 73.91 | 77.27 | 84.09 | 80.43 | 82.22 |
| NEL | 89.67 | 81.68 | 85.49 | 91.94 | 84.65 | 88.14 |
| NETI | 84.21 | 69.57 | 76.19 | 88.37 | 82.61 | 85.39 |
| NEN | 88.89 | 77.78 | 82.96 | 92.86 | 81.25 | 86.67 |
| NEM | 88.28 | 77.40 | 82.48 | 90.85 | 88.36 | 89.58 |
| NETE | 87.00 | 83.12 | 85.02 | 88.60 | 86.62 | 87.60 |

NEs retrieved by the system (correct plus incorrect) and *t* is the total number of NEs in the test data. Using language independent features, we obtained 89.26%, 82.99% and 86.01% precision, recall and F-measure respectively. Then using language independent and dependent features, we obtained 91.20%, 87.17% and 89.13% precision, recall and F-measure respectively. NE tags specific results is shown in Table-3.

**Table 4.** Comparative Evaluation Results.

| **Model** | **F-Measure** |
|---|---|
| Baseline ([23]) | 12.30% |
| Karthik et al., 2008 ([20]) | 40.63% |
| Ekbal et al., 2008a ([17]) | 59.39% |
| Saha et al., 2008 ([10]) | 65.95% |
| Ekbal and Bandyopadhyay, 2010 ([24]) | 84.15% |
| MIRA | **89.13%** |

### 5.3 Comparisons with Existing Systems

The existing Bengali NER systems reported in Table-4 used the same corpus and evaluation metrics as described in this work; i.e., NERSSEAL shared task data and evaluation metrics. The obtained results confirms that the proposed system outperforms the existing models based on CRF, ME, HMM and the best performing existing SVM-based system by 4.98%. The reasons behind the superior performance of the proposed system are the better optimization technique of MIRA and its ability to handle the overlapping features efficiently than the existing systems. Basically MIRA does not explicitly optimize any function, so there is no involvement of probabilistic interpretation. Due to unavailability of experimental data reported in [13, 14, 15] and [18], we were unable to compare this work with those systems.

## 6 Conclusions

This paper presents a system based on MIRA for an IL namely Bengali using both language-independent and language-dependent features. The results show that the proposed system outperforms the existing systems based on CRF, ME, HMM and SVM. But this system was unable to identify NEA (i.e., Abbreviation) properly due to our assumption that short words are rarely NEs which is not true in this case. Post-processing with heuristic patterns may be applied for that. As MIRA has been applied to neither Bengali nor other ILs, so besides improving accuracy one of the notable contributions of this work is to incorporate MIRA in the NER task of one of the ILs. MIRA may be used for other ILs to enhance the performance of state-of-the-art NER systems.

The performance of this work may be enhanced further by applying post-processing with a set of heuristics and Ensemble approaches. Also one of the extension of MIRA, e.g., AdaGrad may be used to improve further performance of NER systems.

## Acknowledgements

We acknowledge the support of the Department of Electronics and Information Technology (DeitY), Ministry of Communications and Information Technology (MCIT), Government of India funded project *"CLIA System Phase II"*.

## References

1. Bandyopadhyay, S.: Multilingual Named Entity Recognition. In: Proceedings of the IJCNLP-08 workshop on NER for South and South East Asian Languages, Hyderabad, India. (2008)
2. Ralph, G.: The New York University System MUC-6 or Where's the syntax?. In: Proceedings of Message Understanding Conference. (1995)
3. McDonald, D.: Internal and external evidence in the identification and semantic categorization of proper names. B. Boguraev and J. Pustejovsky, editors, Corpus Processing for Lexical Acquisition, pp. 21–39. (1996)
4. Takahiro, W., Gaizauskas, R., Wilks, Y.: Evaluation of an algorithm for the recognition and classification of proper names. In: Proceedings of COLING. (1996)
5. Hewavitharana, S., Vogel, S.: Extracting parallel phrases from comparable data. In: Proceedings of the Workshop on Building and Using Comparable Corpora, ACL, pp. 61-–68. Portland, Oregon. (2011)
6. Bikel, D. M.,Scott, M., Richard, S., Ralph, S.: Nymble: A High Performance Learning Name-finder. In: Proceedings of Applied Natural Language Processing, pp. 194-–201. Hyderabad, India. (1997)
7. Wei, L., Andrew, M.: Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction. ACM Transactions on Computational Logic. (2004)

8. Yamada Hiroyasu, Taku Kudo and Yuji Matsumoto: Japanese Named Entity Extraction using Support Vector Machine.Transactions of IPSJ, Vol. 43, No. 1, pp. 44–53. (2002)
9. Andrew, B.: A Maximum Entropy Approach to Named Entity Recognition. Ph.D. Thesis, New York University. (1999)
10. Saha, S. K., Chatterji, S., Dantapat, S., Sarkar, S., Mitra, P.: A Hybrid Approach for Named Entity Recognition in Indian Languages. In: NERSSEAL-IJCNLP-08, pp. 17–24. Hyderabad, India. (2008)
11. Sharma, P., Sharma, U., Kalita, J.: Named Entity Recognition: A Survey for the Indian Languages. Parsing in Indian Languages, pp. 35–39. (2011)
12. Ekbal, A.,Haque, R.,Das, A.,Bandyopadhyay, S.: Language Independent Named Entity Recognition in Indian Languages. In: Proceedings of the NERSSEAL-IJCNLP-08, pp. 33–40. Hyderabad, India. (2008)
13. Ekbal, A. and Saha S.: Weighted Vote Based Classifier Ensemble Selection Using Genetic Algorithm for Named Entity Recognition. In: 15th International Conference on Applications of Natural Language to Information Systems (NLDB 2010), pp. 256–267. Cardiff, UK. (2010)
14. Ekbal, A. and Saha S.: Classifier Ensemble using Multiobjective Optimization for Named Entity Recognition. In: European Conference on Artificial Intelligence (ECAI 2010), pp. 783–788. Lisbon, Portugal. (2010)
15. Ekbal, A. and Saha S.: Maximum Entropy Classifier Ensembling using Genetic Algorithm for NER in Bengali. In: International Conference on Language Resources and Evaluation (LREC 2010), Malta. (2010)
16. Ekbal, A., Bandyopadhyay, S.: Maximum Entropy Approach for Named Entity Recognition in Bengali. In: Proceedings of International Symposium on Natural Language Processing (SNLP-07), pp. 1–6. Thailand. (2007)
17. Ekbal, A., Bandyopadhyay, S.: Bengali Named Entity Recognition using Support Vector Machine.In: NERSSEAL-IJCNLP-08, pp. 51–58. Hyderabad, India. (2008)
18. Ekbal, A., Bandyopadhyay, S.: Voted NER System using Appropriate Unlabeled Data. In: Named Entities Workshop: Shared Task on Transliteration (NEWS 2009), ACL-IJCNLP, pp. 202–210. Singapore. (2009)
19. Chaudhuri, B.,Bhattacharya, S.: An Experiment on Automatic Detection of Named Entities in Bangla. In: NERSSEAL-IJCNLP-08, pp. 75–82. Hyderabad, India. (2008)
20. Gali, K., Surana, H., Vaidya, A., Shishtla, P., Sharma, D. M.: Aggregating Machine Learning and Rule Based Heuristics for Named Entity Recognition.In: NERSSEAL-IJCNLP-08, pp. 25–32. Hyderabad, India. (2008)
21. Ganchev, K., Pereira, F., Mandel, M., Carroll, S., WhiteCrammer, P., Singer, Y.: Semi-automated named entity annotation. In: Proceedings of the linguistic annotation workshop. ACL, pp. 53–56. (2007)
22. Crammer, K.,Singer, Y. : Ultraconservative Online Algorithms for Multiclass Problems. Journal of Machine Learning Research, pp. 951–991. (2003)
23. Singh, A. K.: Named Entity Recognition for South and South East Asian Languages: Taking Stock. In: NERSSEAL-IJCNLP-08. Hyderabad, India. (2008)
24. 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)