Language Independent Named Entity Recognition in Indian Languages

Asif Ekbal, Rejwanul Haque, Amitava Das, Venkateswarlu Poka and Sivaji Bandyopadhyay
Department of Computer Science and Engineering
Jadavpur University
Kolkata-700032, India
asif.ekbal@gmail.com, rejwanul@gmail.com, amit_santu_kuntal@yahoo.com, venkat.ju@gmail.com and sivaji_cse_ju@yahoo.com

Abstract

This paper reports about the development of a Named Entity Recognition (NER) system for South and South East Asian languages, particularly for Bengali, Hindi, Telugu, Oriya and Urdu as part of the IJCNLP-08 NER Shared Task[1]. We have used the statistical Conditional Random Fields (CRFs). The system makes use of the different contextual information of the words along with the variety of features that are helpful in predicting the various named entity (NE) classes. The system uses both the language independent as well as language dependent features. The language independent features are applicable for all the languages. The language dependent features have been used for Bengali and Hindi only. One of the difficult tasks of IJCNLP-08 NER Shared task was to identify the nested named entities (NEs) though only the type of the maximal NEs were given. To identify nested NEs, we have used rules that are applicable for all the five languages. In addition to these rules, gazetteer lists have been used for Bengali and Hindi. The system has been trained with Bengali (122,467 tokens), Hindi (502,974 tokens), Telugu (64,026 tokens), Oriya (93,173 tokens) and Urdu (35,447 tokens) data. The system has been tested with the 30,505 tokens of Bengali, 38,708 tokens of Hindi, 6,356 tokens of Telugu, 24,640 tokens of Oriya and 3,782 tokens of Urdu. Evaluation results have demonstrated the highest maximal F-measure of 53.36%, nested F-measure of 53.46% and lexical F-measure of 59.39% for Bengali.

1 Introduction

Named Entity Recognition (NER) is an important tool in almost all Natural Language Processing (NLP) application areas. Proper identification and classification of named entities are very crucial and pose a very big challenge to the NLP researchers. The level of ambiguity in named entity recognition (NER) makes it difficult to attain human performance.

NER has drawn more and more attention from the named entity (NE) tasks (Chinchor 95; Chinchor 98) in Message Understanding Conferences (MUCs) [MUC6; MUC7]. The problem of correct identification of named entities is specifically addressed and benchmarked by the developers of Information Extraction System, such as the GATE system (Cunningham, 2001). NER also finds application in question-answering systems (Maldovan et al., 2002) and machine translation (Babych and Hartley, 2003).

The current trend in NER is to use the machine-learning approach, which is more attractive in that it is trainable and adoptable and the maintenance of a machine-learning system is much cheaper than that of a rule-based one. The representative machine-learning approaches used in NER are HMM (BBN's IdentiFinder in (Bikel, 1999)), Maximum Entropy

[1]http://ltrc.iiit.ac.in/ner-ssea-08

Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 33–40,
Hyderabad, India, January 2008.

(New York University's MENE in (Borthwick, 1999)), Decision Tree (New York University's system in (Sekine 1998), SRA's system in (Bennet, 1997) and Conditional Random Fields (CRFs) (Lafferty et al., 2001; McCallum and Li, 2003).

There is no concept of capitalization in Indian languages (ILs) like English and this fact makes the NER task more difficult and challenging in ILs. There has been very little work in the area of NER in Indian languages. In Indian languages particularly in Bengali, the work in NER can be found in (Ekbal and Bandyopadhyay, 2007a) and (Ekbal and Bandyopadhyay, 2007b). These two systems are based on the pattern directed shallow parsing approach. An HMM-based NER in Bengali can be found in (Ekbal et al., 2007c). Other than Bengali, the work on NER can be found in (Li and McCallum, 2004) for Hindi. This system is based on CRF.

In this paper, we have reported a named entity recognition system for the south and south east Asian languages, particularly for Bengali, Hindi, Telugu, Oriya and Urdu. Bengali is the seventh popular language in the world, second in India and the national language of Bangladesh. Hindi is the third popular language in the world and the national language of India. Telugu is one of the popular languages and predominantly spoken in the southern part of India. Oriya and Urdu are the other two popular languages of India and widely used in the eastern and the northern part, respectively. The statistical Conditional Random Field (CRF) model has been used to develop the system, as it is more efficient than HMM to deal with the non-independent and diverse overlapping features of the highly inflective Indian languages. We have used a fine-grained named entity tagset[2], defined as part of the IJCNLP-08 NER Shared Task for SSEA. The system makes use of the different contextual information of the words along with the variety of orthographic word level features that are helpful in predicting the various named entity classes. In this work, we have considered language independent features as well as the language dependent features. Language independent features include the contextual words, prefix and suffix information of all the words in the training corpus, several digit features depending upon the presence and/or the number of digits in a token and the frequency features of the words. The system considers linguistic features particularly for Bengali and Hindi. Linguistic features of Bengali include the set of known suffixes that may appear with named entities, clue words that help in predicating the location and organization names, words that help to recognize measurement expressions, designation words that help in identifying person names, the various gazetteer lists like the first names, middle names, last names, location names and organization names. As part of linguistic features for Hindi, the system uses only the lists of first names, middle names and last names along with the list of words that helps to recognize measurements. No linguistic features have been considered for Telugu, Oriya and Urdu. It has been observed from the evaluation results that the use of linguistic features improves the performance of the system. A number of experiments have been carried out to find out the best-suited set of features for named entity recognition in Bengali, Hindi, Telugu, Oriya and Urdu.

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

2 Conditional Random Fields

Conditional Random Fields (CRFs) (Lafferty et al., 2001) are undirected graphical models, a special case of which corresponds to conditionally trained probabilistic finite state automata. Being conditionally trained, these CRFs can easily incorporate a large number of arbitrary, non-independent features while still having efficient procedures for non-greedy finite-state inference and training. CRFs have shown success in various sequence modeling tasks including noun phrase segmentation (Sha and Pereira, 2003) and table extraction (Pinto et al., 2003).

CRFs are used to calculate the conditional probability of values on designated output nodes given values on other designated input nodes. The conditional probability of a state sequence $S = < s_1, s_2, \ldots, s_T >$ given an observation sequence $O = < o_1, o_2, \ldots\ldots, o_T)$ is calculated as:

$P_\Lambda(s \mid o) = \frac{1}{Z_o} \exp(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(s_{t-1}, s_t, o, t))$, where $f_k(s_{t-1}, s_t, o, t)$ is a feature function whose weight λ_k is to be learned via training. The values of the feature functions may range between $-\propto \ldots\ldots + \propto$, but typically they are binary. To make all

conditional probabilities sum up to 1, we must calculate the normalization factor, $Z_0 = \sum_s \exp(\sum_{t=1}^{T} \sum_k \lambda_k f_k(s_{t-1}, s_t, o, t))$, which, as in HMMs, can be obtained efficiently by dynamic programming.

To train a CRF, the objective function to be maximized is the penalized log-likelihood of the state sequences given observation sequences:

$$L_\wedge = \sum_{i=1}^{N} \log(P_\wedge(s^{(i)} \mid o^{(i)})) - \sum_k \frac{\lambda_k^2}{2\sigma^2},$$

where, $\{ < o^{(i)}, s^{(i)} > \}$ is the labeled training data. The second sum corresponds to a zero-mean, σ^2 -variance Gaussaian prior over parameters, which facilitates optimization by making the likelihood surface strictly convex. Here, we set parameters λ to maximize the penalized log-likelihood using Limited-memory BFGS (Sha and Pereira, 2003), a quasi-Newton method that is significantly more efficient, and which results in only minor changes in accuracy due to changes in σ .

When applying CRFs to the named entity recognition problem, an obsevation sequence is a token of a sentence or document of text and the state sequence is its corresponding label sequence. While CRFs generally can use real-valued functions, in our experiments maximum of the features are binary. A feature function $f_k(s_{t-1}, s_t, o, t)$ has a value of 0 for most cases and is only set to be 1, when s_{t-1}, s_t are certain states and the observation has certain properties. We have used the C^{++} based OpenNLP CRF++ package[3], a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting /labeling sequential data.

3 Named Entity Recognition in Indian Languages

Named Entity Recognition in Indian languages (ILs) is difficult and challenging as capitalization is not a clue in ILs. The training data were provided for five different Indian languages, namely Bengali, Hindi, Telugu, Oriya and Urdu in *Shakti Standard Format*[4]. The training data in all the languages were annotated with the twelve NE tags, as defined for the IJCNLP-08 NER shared task taget[5]. Only the maximal named entities and not the internal structures of the entities were annotated in the training data. For example, *mahatma gandhi road* was annotated as location and assigned the tag 'NEL' even if *mahatma* and *gandhi* are named entity title person (NETP) and person name (NEP) respectively, according to the IJCNLP-08 shared task tagset. These internal structures of the entities were to be identified during testing. So, *mahatma gandhi road* will be tagged as *mahatma* /NETP *gandhi*/NEP *road*/NEL. The structure of the tagged element using the *SSF* form will be as follows:

```
1       ((      NP      <ne=NEL>
1.1     ((      NP      <ne=NEP>
1.1.1   ((      NP      <ne=NETP>
1.1.1.1 mahatma
        ))
1.1.2   gandhi
        ))
1.2     road
))
```

3.1 Training Data Preparation for CRF

Training data for all the languages required some preprocessing in order to use in the Conditional Random Field framework. The training data is searched for the multiword NEs. Each component of the multiword NE is searched in the training set to find whether it occurs as a single-word NE. The constituent components are then replaced by their NE tags (NE type of the single-word NE). For example, *mahatma gandhi road*/NEL will be tagged as *mahatma*/NETP *gandhi*/NEP *road*/NEL if the internal components are found to appear with these NE tags in the training set. Each component of a multiword NE is also checked whether the component is made up of digits only. If a component is made up digits only, then it is assigned the tag 'NEN'. Various gazetteers for Bengali and Hindi have been also used in order to identify the internal structure of the NEs properly. The list of gazetteers, which have been used in preparing the training data, is shown in Table 1.

The individual components (not occurring as a single-word NE in the training data) of a multiword NE are searched in the gazetteer lists and

[3]http://crfpp.sourceforge.net

[4]http://shiva.iiit.ac.in/SPSAL 2007/ssf.html

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

assigned the appropriate NE tags. Other than NEs are marked with the NNE tags. The procedure is given below:

Gazetteer list	Number of entries
First person name in Bengali	27,842
Last person name in Bengali	5,288
Middle name in Bengali	1,491
Person name designation in Bengali	947
Location name in Bengali	7,870
First person name in Hindi	1,62,881
Last person name in Hindi	3,573
Middle name in Hindi	450
Cardinals in Bengali, Hindi and Telugu	100
Ordinals in Bengali, Hindi and Telugu	65
Month names in Bengali, Hindi and Telugu	24
Weekdays in Bengali, Hindi and Telugu	14
Words that denote measurement in Bengali, Hindi and Telugu	52

Table 1. Gazetteer lists used during training data preparation

Step 1: Search the multiword NE in the training data
Step 2: Extract each component from the multword NE.
Step 3: Check whether the constituent individual component (except the last one) appears in the training data as a single-word NE.
Step 4: If the constituent NE appears in the training data as a single-word NE then
Step 4.1: Assign the NE tag, extracted from the single-word NE, to the component of the multiword NE.
else
Step 4.2: Search the component in the gazetteer lists and assign the appropriate NE tag.
Step 4.2.1: If the component is not found to appear in the gazetteer list then assign the NE tag of the maximal NE to the individual component.

For example, if *mahatma gandhi road* is tagged as NEL, i.e., *mahatma gandhi road*/NEL then each component except the last one (*road*) of this multword NE is searched in the training set to look for it's appearance (Step 3). Gazetteer lists are searched in case the component is not found in the training set (Step 4.2). If the components are found either in the training set or in the gazetteer list, then *mahatma gandhi road*/NEL will be tagged as: *mahatma*/NETP *gandhi*/NEP *road*/NEL.

3.2 Named Entity Features

Feature selection plays a crucial role in CRF framework. Experiments were carried out to find out most suitable features for NE tagging task. The main features for the NER task have been identified based on the different possible combination of available word and tag context. The features also include prefix and suffix for all words. The term prefix/suffix is a sequence of first/last few characters of a word, which may not be a linguistically meaningful prefix/suffix. The use of prefix/suffix information works well for highly inflected languages like the Indian languages. In addition, various gazetteer lists have been developed to use in the NER task particularly for Bengali and Hindi. We have considered different combination from the following set for inspecting the best feature set for the NER task:

F={ $w_{i-m},...,w_{i-1},w_i,w_{i+1},...,w_{i+n}$, |prefix|≤n, |suffix|≤n, previous NE tag, POS tags, First word, Digit information, Gazetteer lists}

Following is the details of the set of features that were applied to the NER task:

• Context word feature: Previous and next words of a particular word might be used as a feature. We have considered the word window of size five, i.e., previous and next two words from the current word for all the languages.

•Word suffix: Word suffix information is helpful to identify NEs. A fixed length word suffix of the current and surrounding words might be treated as feature. In this work, suffixes of length up to three the current word have been considered for all the languages. More helpful approach is to modify the feature as binary feature. Variable length suffixes of a word can be matched with predefined lists of useful suffixes for different classes of NEs. For Bengali, we have considered the different suffixes that may be particularly helpful in detecting person (e.g., *-babu, -da, -di* etc.).

•Word prefix: Prefix information of a word is also helpful. A fixed length prefix of the current and the surrounding words might be treated as features. Here, the prefixes of length up to three have been considered for all the language.

•Rare word: The lists of most frequently occurring words in the training sets have been calculated for all the five languages. The words that occur more than 10 times are considered as the frequently occurring words in Bengali and Hindi. For Telugu, Oriya and Urdu, the cutoff frequency was chosen to be 5. Now, a binary feature 'RareWord' is defined as: If current word is found to appear in the frequent word list then it is set to 1; otherwise, set to 0.

•First word: If the current token is the first word of a sentence, then this feature is set to 1. Otherwise, it is set to 0.

•Contains digit: For a token, if it contains digit(s) then the feature 'ContainsDigit' is set to 1. This feature is helpful for identifying the numbers.

•Made up of four digits: For a token if all the characters are digits and having 4 digits then the feature 'FourDigit' is set to 1. This is helpful in identifying the time (e.g., *2007sal*) and numerical (e.g., *2007*) expressions.

•Made up of two digits: For a token if all the characters are digits and having 2 digits then the feature 'TwoDigit' is set to 1. This is helpful for identifying the time expressions (e.g., *12 ta, 8 am, 9 pm*) in general.

•Contains digits and comma: For a token, if it contains digits and commas then the feature 'ContainsDigitsAndComma' is set to 1. This feature is helpful in identifying named entity measurement expressions (e.g., *120,45,330 taka*) and numerical numbers (e.g., *120,45,330*)

•Contains digits and slash: If the token contains digits and slash then the feature 'ContainsDigitAndslash' is set to 1. This helps in identifying time expressions (e.g., 15/8/2007).

•Contains digits and hyphen: If the token contains digits and hyphen then the feature 'ContainsDigitsAndHyphen' is set to 1. This is helpful for the identification of time expressions (e.g., 15-8-2007).

•Contains digits and period: If the token contains digits and periods then the feature 'ContainsDigitsAndPeriod' is set to 1. This helps to recognize numerical quantities (e.g., *120453.35*) and measurements (e.g., *120453.35 taka*).

•Contains digits and percentage: If the token contains digits and percentage symbol then the feature 'ContainsDigitsAndPercentage' is set to 1. This helps to recognize measurements (e.g., *120%*).

•Named Entity Information: The NE tag of the previous word is also considered as the feature, i.e., the combination of the current and the previous output token has been considered. This is the only dynamic feature in the experiment.

•Gazetteer Lists: Various gazetteer lists have been created from a tagged Bengali news corpus (Ekbal and Bandyopadhyay, 2007d) for Bengali. The first, last and middle names of person for Hindi have been created from the election commission data[6]. The person name collections had to be processed in order to use it in the CRF framework. The simplest approach of using these gazetteers is to compare the current word with the lists and make decisions. But this approach is not good, as it can't resolve ambiguity. So, it is better to use these lists as the features of the CRF. If the current token is in a particular list, then the corresponding feature is set to 1 for the current/previous/next token; otherwise, set to 0. The list of gazetteers is shown in Table 2.

3.3 Nested Named Entity Identification

One of the important tasks of the IJCNLP-NER shared task was to identify the internal named entities within the maximal NEs. In the training data, only the type of the maximal NEs were given. In order to identify the internal NEs during testing, we have defined some rules. After testing the unannotated test data with the CRF based NER system, it is searched to find the sequence of NE tags. The last NE tag in the sequence is assigned as the NE tag of the maximal NE. The NE tags of the constituent NEs may either be changed or may not be changed. The NE tags are changed with the help of rules and various gazetteer lists. We identified NEM (Named entity measurement), NETI (Named entity time expressions), NEO (Named entity organization names), NEP (Named entity person names) and NEL (Named entity locations) to be the potential NE tags, where nesting could occur. A NEM expression may contain NEN, an NETI may contain NEN, an NEO may contain NEP/ NEL, an NEL may contain NEP/NETP/NED and an NEP may contain NEL expressions. The nested

[6] http://www.eci.gov.in/DevForum/Fullname.asp

NEN tags could be identified by simply checking whether it contains digits only and checking the lists of cardinal and ordinal numbers.

Gazetteer	Number of entries	Feature Descriptions
Designation words in Bengali	947	'Designation' set to 1, otherwise 0
Organization names in Bengali	2, 225	'Organization' set to 1, otherwise 0.
Organization suffixes in Bengali	94	'OrgSuffix' set to 1, otherwise 0
Person prefix for Bengali	245	'PersonPrefix' set to 1, otherwise set to 0
First person names in Bengali	27,842	'FirstName' set to 1, otherwise 0
Middle names in Bengali	1,491	'MiddleName' set to 1, otherwise 0
Surnames in Bengali	5,288	'SurName' set to 1, otherwise 0
Common location word in Bengali	75	'CommonLocation' set 1, otherwise 0
Action verb in Bengali	215	'ActionVerb' set to 1, otherwise 0
First person names in Hindi	1,62,881	'FirstName' set to 1, otherwise 0
Middle person names in Hindi	450	'MiddleName' set to 1, otherwise 0
Last person names in Hindi	3,573	'SurName' set to 1, otherwise 0
Location names in Bengali	7,870	'LocationName' set to 1, otherwise 0
Week days in Bengali, Hindi and Telugu	14	'WeekDay' set to 1, otherwise 0
Month names in Bengali, Hindi and Telugu	24	'MonthName' set to 1, otherwise 0
Measurements in Bengali, Hindi and Telugu	52	'Measurement' set to 1, otherwise 0.

Table 2. Named entity gazetteer list

The procedure for identifying the nested NEs are shown below:
Step1: Test the unannotated test set.
Step 2: Look for the sequence of NE tags.
Step 3: All the words in the sequence will belong to a maximal NE.
Step 4: Assign the last NE tag in the sequence to the maximal NE.
Step 5: The test set is searched to look whether each component word appears with a NE tag.
Step 6: Assign the particular NE tag to the component if it appears in the test set with that NE tag. Otherwise, search the gazetteer lists as shown in Tables 1-2 to assign the tag.

4 Evaluation

The evaluation measures used for all the five languages are precision, recall and F-measure. These measures are calculated in three different ways:

(i). Maximal matches: The largest possibles named entities are matched with the reference data.

(ii). Nested matches: The largest possible as well as nested named entities are matched.

(iii). Maximal lexical item matches: The lexical items inside the largest possible named entities are matched.

(iv). Nested lexical item matches: The lexical items inside the largest possible as well as nested named entities are matched.

5 Experimental Results

The CRF based NER system has been trained and tested with five different Indian languages namely, Bengali, Hindi, Telugu, Oriya and Urdu data. The training and test sets statistics are presented in Table 3. Results of evaluation as explained in the previous section are shown in Table 4. The F-measures for the nested lexical match are also shown individually for each named entity tag separately in Table 5.

Experimental results of Table 4 show that the CRF based NER system performs best for Bengali with maximal F-measure of 55.36%, nested F-measure of 61.46% and lexical F-measure 59.39%. The system has demonstrated the F-measures of 35.37%, 36.75% and 33.12%, respectively for maximal, nested and lexical matches. The system has shown promising precision values for Hindi. But due to the low recall values, the F-measures get reduced. The large difference between the recall and precision values in the evaluation results of Hindi indicates that the system is not able to retrieve a significant number of NEs from the test

data. In comparison to Hindi, the precision values are low and the recall values are high for Bengali. It can be decided from the evaluation results that system retrieves more NEs in Bengali than Hindi but involves more errors. The lack of features in Oriya, Telugu and Urdu might be the reason behind their poor performance.

Language	Number of tokens in the training set	Number of tokens in the test set
Bengali	122,467	30,505
Hindi	502,974	38,708
Telugu	64,026	6,356
Oriya	93,173	24,640
Urdu	35,447	3,782

Table 3: Training and Test Sets Statistics

Tag	Bengali	Hindi	Oriya	Telugu	Urdu
NEP	85.68	21.43	43.76	1.9	7.69
NED	35.9	38.70	NF	NF	NF
NEO	52.53	NF	5.60	NF	22.02
NEA	26.92	30.77	NF	NF	NF
NEB	NF	NF	NF	NF	NF
NETP	61.44	NF	12.55	NF	NF
NETO	45.98	NF	NF	NF	NF
NEL	80.00	22.70	31.49	0.73	50.14
NETI	53.43	49.60	27.08	7.64	49.28
NEN	30.12	85.40	9.19	9.16	NF
NEM	79.08	36.64	7.56	NF	79.27
NETE	18.06	1.64	NF	5.74	NF

Table 4. Evaluation for Specific NE Tags (F-Measures for nested lexical match) [NF: Nothing found]

Experimental results of Table 5 show the F-measures for the nested lexical item matches for individual NE tags. For Bengali, the system has shown reasonably high F-measures for NEP, NEL and NEM tags and medium F-measures for NETP, NETI, NEO and NETO tags. The overall F-measures in Bengali might have reduced due to relatively poor F-measures for NETE, NEN, NEA and NED tags. For Hindi, the highest F-measure obtained is 85.4% for NEN tag followed by NETI, NED, NEM, NEA, NEL and NEP tags. In some cases, the system has shown better F-measures for Hindi than Bengali also. The system has performed better for NEN, NED and NEA tags in Hindi than all other languages.

6 Conclusion

We have developed a named entity recognition system using Conditional Random Fields for the five different Indian languages, namely Bengali, Hindi, Telugu, Oriya and Urdu. We have considered the contextual window of size five, prefix and suffix of length upto three of the current word, NE information of the previous word, different digit features and the frequently occurring word lists. The system also uses linguistic features extracted from the various gazetteer lists for Bengali and Hindi. Evaluation results show that the system performs best for Bengali. The performance of the system for Bengali can further be improved by including the part of speech (POS) information of the current and/or the surrounding word(s). The performance of the system for other languages can be improved with the use of different linguistic features as like Bengali.

The system did not perform as expected due to the problems faced during evaluation regarding the tokenization. We have tested the system for Bengali with 10-fold cross validation and obtained impressive results.

References

Babych, Bogdan, A. Hartley. Improving machine translation quality with automatic named entity recognition. In *Proceedings of EAMT/EACL 2003 Workshop on MT and other language technology tools*, 1-8, Hungary.

Bennet, Scott W.; C. Aone; C. Lovell. 1997. Learning to Tag Multilingual Texts Through Observation. In *Proceedings of EMNLP*, 109-116, Rhode Island.

Bikel, Daniel M., R. Schwartz, Ralph M. Weischedel. 1999. An Algorithm that Learns What's in Name. *Machine Learning (Special Issue on NLP)*, 1-20.

Bothwick, Andrew. 1999. A Maximum Entropy Approach to Named Entity Recognition. *Ph.D. Thesis*, New York University.

Chinchor, Nancy. 1995. MUC-6 Named Entity Task Definition (Version 2.1). *MUC-6*, Columbia, Maryland.

Measure→	Precision			Recall			F-measure		
Language ↓	P_m	P_n	P_l	R_m	R_n	R_l	F_m	F_n	F_l
Bengali	51.63	47.74	52.90	59.60	61.46	67.71	55.36	61.46	59.39
Hindi	71.05	76.08	80.59	23.54	24.23	20.84	35.37	36.75	33.12
Oriya	27.12	27.18	50.40	12.88	10.53	20.07	17.47	15.18	28.71
Telugu	1.70	2.70	8.10	0.538	0.539	3.34	0.827	0.902	4.749
Urdu	49.16	48.93	54.45	21.95	20.15	26.36	30.35	28.55	35.52
M: Maximal, *n*: Nested, *l*: Lexical									

Table 5. Evaluation of the Five Languages

Chinchor, Nancy. 1998. MUC-7 Named Entity Task Definition (Version 3.5). *MUC-7*. Fairfax, Virginia.

Cunningham, H. 2001. GATE: A general architecture for text engineering. *Comput. Humanit*. (36), 223-254.

Ekbal, Asif, and S. Bandyopadhyay. 2007a. Pattern Based Bootstrapping Method for Named Entity Recognition. In *Proceedings of 6th International Conference on Advances in Pattern Recognition*, Kolkata, India, 349-355.

Ekbal, Asif, and S. Bandyopadhyay. 2007b. Lexical Pattern Learning from Corpus Data for Named Entity Recognition. In *Proceedings of the 5th International Conference on Natural Language Processing*, Hyderabad, India, 123-128.

Ekbal, Asif, Naskar, Sudip and S. Bandyopadhyay. 2007c. Named Entity Recognition and Transliteration in Bengali. *Named Entities: Recognition, Classification and Use, Special Issue of Lingvisticae Investigationes Journal*, 30:1 (2007), 95-114.

Ekbal, Asif, and S. Bandyopadhyay. 2007d. A Web-based Bengali News Corpus for Named Entity Recognition. *Language Resources and Evaluation Journal (Accepted)*

Lafferty, J., McCallum, A., and Pereira, F. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of 18th International Conference on Machine learning*.

Li, Wei and Andrew McCallum. 2003. Rapid Development of Hindi Named Entity Recognition Using Conditional Random Fields and Feature Inductions, *ACM TALIP*, 2(3), (2003), 290-294.

McCallum, A.; W. Li. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-Enhanced Lexicons. In *Proceedings CoNLL-03*, Edmanton, Canada.

Moldovan, Dan I., Sanda M. Harabagiu, Roxana Girju, P. Morarescu, V. F. Lacatusu, A. Novischi, A. Badulescu, O. Bolohan. 2002. LCC Tools for Question Answering. In *Proceedings of the TREC*, Maryland, 1-10.

Pinto, D., McCallum, A., Wei, X., and Croft, W. B. 2003. Table extraction using conditional random fields. In *Proceedings of SIGIR 03 Conference*, Toronto, Canada.

Sekine, Satoshi. 1998. Description of the Japanese NE System Used for MET-2, *MUC-7*, Fairfax, Virginia.

Sha, F. and Pereira, F. 2003. Shallow parsing with conditional random fields. In *Proceedings of Human Language Technology*, NAACL.