Linguistic **I**ssues in **L**anguage **T**echnology – **LiLT**
Volume 2, Issue 1 November, 2009

A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi

Asif Ekbal
Sivaji Bandyopadhyay

Published by CSLI Publications

A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi

Asif Ekbal, *Department of Computational Linguistics, University of Heidelberg, 69120 Heidelberg, Germany*
Email: ekbal@cl.uni-heidelberg.de, asif.ekbal@gmail.com
Sivaji Bandyopadhyay, *Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India*
Email: sivaji_cse_ju@yahoo.com, sbandyopadhyay@cse.jdvu.ac.in

Abstract:This paper describes the development of Named Entity Recognition (NER) systems for two leading Indian languages, namely Bengali and Hindi using the Conditional Random Field (CRF) framework. The system makes use of different types of contextual information along with a variety of features that are helpful in predicting the different named entity (NE) classes. This set of features includes language independent as well as language dependent components. We have used the annotated corpora of 122,467 tokens for Bengali and 502,974 tokens for Hindi tagged with a tag set [1] *of twelve different NE classes, defined as part of the IJCNLP-08 NER Shared Task for South and South East Asian Languages (SSEAL)* [2]*. We have considered only the tags that denote person names, location names, organization names, number expressions, time expressions and measurement expressions. A number of*

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

LiLT Volume 2, Issue 1, November 2009.
A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi.

experiments have been carried out in order to find out the most suitable features for NER in Bengali and Hindi. The system has been tested with the gold standard test sets of 35K for Bengali and 50K tokens for Hindi. Evaluation results in overall f-score values of 81.15% for Bengali and 78.29% for Hindi for the test sets. 10-fold cross validation tests yield f-score values of 83.89% for Bengali and 80.93% for Hindi. ANOVA analysis is performed to show that the performance improvement due to the use of language dependent features is statistically significant.

Keywords:Named Entity, Named Entity Recognition, Conditional Random Field, Bengali, Hindi.

1 Introduction

Named Entity Recognition (NER) is an important tool in almost all Natural Language Processing (NLP) application areas such as Information Retrieval, Information Extraction, Machine Translation, Question Answering and Automatic Summarization. The objective of NER is to identify and classify every word/term in a document into some predefined categories like person name, location name, organization name, miscellaneous name (date, time, percentage and monetary expressions etc.) and "none-of-the-above". The challenge in detection of named entities (NEs) is that such expressions are hard to analyze using rule-based NLP because they belong to the open class of expressions, i.e., there is an infinite variety and new expressions are constantly being invented.

The level of ambiguity in NER makes it difficult to attain human performance. There are two kinds of evidence that can be used in NER to solve the ambiguity, robustness and portability problems. The first is the internal evidence found within the word and/or word string itself, while the second is the external evidence gathered from its context. NER has drawn more and more attention from NLP researchers since the last decade (Chinchor 1995,Chinchor 1998) in Message Understanding Conferences (MUCs)(Chinchor 1995,Chinchor 1998). Correct identification of NEs is specifically addressed and benchmarked by the developers of Information Extraction System, such as the GATE system (Cunningham 2002). NER also finds applications in question-answering systems (Moldovan et al. 2002) and machine translation (Babych and Hartley 2003).

Previous approaches have typically used manually constructed finite state patterns, which attempt to match against a sequence of words in much the same way as a general regular expression matcher. Typical systems are University Of Sheffield's LaSIE-II (Humphreys et al. 1998), ISOQuest's NetOwl (Aone et al. 1998) and University Of Edinburgh's LTG (Mikheev et al. 1998, Mikheev et al. 1999) for English NER. These systems are mainly rule-based. However, rule-based approaches lack the ability of coping with the problems of robustness and portability. Each new source of text requires significant tweaking of rules to maintain optimal performance and the maintenance costs can be quite steep.

Nowadays, machine-learning (ML) approaches are popularly used in NER because these are easily trainable, adaptable to different domains and languages and their maintenance is less expensive (Zhou and Su 2002). Representative machine-learning approaches used in NER are Hidden Markov Model (HMM)(BBN's IdentiFinder in (Miller et al. 1998, Bikel et al. 1999)), Maximum Entropy (ME) systems (New York University's MENE in (Borthwick 1999, Borthwick et al. 1998)), Decision Tree (New York University's system in (Sekine 1998) and SRA's system in (Bennet et al. 1997)) and CRF (McCallum and Li 2003, Lafferty et al. 2001). NER can also be treated as a tagging problem, where each word in a sentence is assigned a label indicating whether it is part of a NE and the entity type. Thus methods used for part of speech (POS) tagging can also be used for NER. The papers from the CoNLL-2002 shared task, which used such methods (Malouf 2002, Burger et al. 2002) show results significantly lower than the best system (Carrears et al. 2002). However, Zhou and Su (2002) have reported state of the art results on the MUC-6 and MUC-7 data using an HMM-based tagger.

1.1 Existing Works on NER in Indian Languages

India is a multilingual country with great cultural diversities. In the area of NER work involving Indian languages has started only very recently. Named Entity (NE) identification in Indian languages is difficult and challenging as:

1. Unlike English and most of the European languages, Indian languages lack the capitalization information that plays a very important role to identify NEs in those languages;
2. Indian person names are more diverse compared to these of most other languages and a lot of them can be found in the dictionary as common nouns;
3. Indian languages are highly inflected and provide rich and challenging sets of linguistic and statistical features resulting in long and complex word forms;
4. Indian languages have relatively free word order;
5. Bengali and Hindi, like other Indian languages, are also resource poor languages-annotated corpora, name dictionaries, good mor-

phological analyzers, POS taggers etc. are not yet available in the required quantity and quality;

6. Although Indian languages have a very old and rich literary history, technological developments are recent;
7. Web sources for name lists are available in English, but such lists are not available in Indian languages forcing the use of transliteration.

A pattern-directed shallow parsing approach for NER in Bengali has been described in Ekbal and Bandyopadhyay (2007a). The paper reports on two different NER models, one using lexical contextual patterns and the other using linguistic features along with the same set of lexical contextual patterns. A HMM-based NER system has been described in Ekbal et al. (2007b), where more contextual information has been taken into consideration during the emission probabilities and NE suffixes have been kept for handling unknown words. More recent works in the area of Bengali NER can be found in Ekbal et al. (2008), and Ekbal and Bandyopadhyay (2008a) with a CRF, and a SVM approach, respectively. These systems were developed with the help of different contextual and orthographic word-level features along with a variety of features extracted from the gazetteers.

The work on Hindi NER can be found in Li and McCallum (2004) with a CRF approach that uses a feature induction technique to automatically construct the features that most increase the conditional likelihood. A language independent method for Hindi NER has been reported in Cucerzon and Yarowsky (1999). Saha et al. (2008)reported a ME based system with a hybrid feature set that includes statistical as well as linguistic features. A MEMM-based system has been reported in Kumar and Bhattacharyya (2006). Various systems of NER in Indian languages using different approaches have been reported as part of the IJCNLP-08 NER Shared Task on South and South East Asian Languages (NERSSEAL)[3]. As part of this shared task, Gali et al. (2008) described a CRF-based system that uses post-processing with some heuristics or rules for Bengali, Hindi, Oriya, Telugu, and Urdu. Another CRF-based system has been described in Kumar and Kiran (2008), where it has been shown that a hybrid HMM model can perform better than CRF. In the first phase, HMM models are trained on the training corpus and are used to tag the test data. The first layer is purely statistical and the second layer is a pure rule-based. In order to extend the tool to any other Indian language they formulated the rules

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

in the second layer. They tested their system for Bengali, Hindi, Oriya, Telugu and Urdu.

Srikanth and Murthy (2008) developed a NER system for Telugu and tested it on several data sets from the Eenaadu and Andhra Prabha newspaper corpora. They obtained an overall f-measure between 80-97% with person, location and organization tags. For Tamil, a CRF-based NER system has been presented in Vijayakrishna and Sobha (2008) for the tourism domain. This approach can take care of morphological inflections of NEs and can handle nested tagging with a hierarchical tag set containing 106 tags. Shishtla et al. (2008) developed a CRF-based system for English, Telugu and Hindi. They suggested that a character n-gram based approach is more effective than word based models. They described the features they used and their experiments to increase the recall of NER system.

In this paper, we describe a NER systems for two leading Indian languages, Bengali and Hindi. In terms of native speakers, Bengali is the seventh most spoken language in the world, it is the second in India as well as the national language of Bangladesh. Hindi is the third most spoken language in the world and the national language of India. A CRF model has been used to develop the NER systems because of its efficiency in dealing with the non-independent, diverse and overlapping features of the highly inflective Indian languages. We have used the IJCNLP-08 NERSSEAL Shared Task data that was originally annotated with a fine-grained NE tag set of twelve tags. We considered only the tags that denote person names (NEP), location names (NEL), organization names (NEO), number expressions (NEN), time expressions (NETI) and measurement expressions (NEM). The NEN, NETI and NEM tags are mapped to the *Miscellaneous* NE tag that denotes miscellaneous entities. The system makes use of different types of contextual information of the words along with a variety of orthographic word level features that are helpful in predicting the various NE classes. We have taken into consideration both language independent and language dependent features. Language dependent features have been extracted from language specific resources such as gazetteers. The POS information has been used in two different ways. Initially, language independent POS taggers have been developed and used as language independent features for both Bengali and Hindi. Then, a language dependent POS tagger has been developed for Bengali using several language specific resources and using language dependent features. It has been observed that the language specific features play a crucial role in improving the system's performance. ANOVA analysis shows that the performance improvement with the language dependent features is statistically significant in each of the languages. We have also carried out a number of experiments to find out the best-suited set of features for NER for each of the languages. In the IJCNLP-08 NERSSEAL shared task, the major challenge was to identify the NE classes of the constituent parts, i.e., of the nested NEs. Here, we do not concentrate on the recognition of the nested NEs. We are interested in the classification of NEs with their maximal type only. The work reported in this paper differs from that of the work reported in Ekbal et al. (2008) in following ways:

1. This work deals with NER for two different Indian languages, Bengali and Hindi;
2. More NE features are incorporated into the system;
3. The system has been developed in two ways:
   - With language independent features that are applicable to both Bengali and Hindi.
   - With language dependent (applicable to Bengali and/or Hindi) features;
4. Performance of the reported NER system has been compared with three existing Bengali NER systems;
5. An ANOVA statistical analysis is performed to show that the performance improvement using language specific features is statistically significant;
6. The impact of the language dependent features on the evaluation results for each NE tag has been demonstrated.

The rest of this paper is organized as follows. The NER problems in Indian languages along with the NE tag set and CRF framework are described in Section 2. Section 3 describes the details of the NE features that are applicable to almost all the languages and the language specific features for Bengali and Hindi. Detailed evaluation results of the system for the development sets, test sets and cross-validation tests are presented in Section 4. Section 5 concludes the paper.

2 Named Entity Recognition in Bengali and Hindi

Applying stochastic models to the NER problem requires large amounts of annotated data in order to achieve reasonable performance. Stochastic models have been applied successfully to English, German and other European languages for which large sets of labeled data are available. The problem remains difficult for Indian languages (ILs) due to the lack of such large annotated corpora.

Simple HMMs do not work well when small amounts of labeled data are used to estimate the model parameters. Incorporating diverse features in an HMM based NE tagger is difficult and complicates the smoothing typically used in such taggers. In contrast, a ME (Borthwick 1999), CRF (Lafferty et al. 2001) or SVM (Yamada et al. 2001) based method can deal with the diverse and overlapping features of the Indian languages. In this work, we have used CRF to identify and classify NEs in Bengali and Hindi.

2.1 Named Entity Tag Set

We have used the IJCNLP-08 NERSSEAL shared task data that were tagged with twelve NE tags. The tag set consists of more tags than the four tags of CoNLL 2003 shared task on NER. The underlying reason for adopting this finer NE tag set is to use the NER system in various NLP applications, particularly in machine translation. The IJCNLP-08 NERSSEAL shared task tag set is shown in Table 1. One important aspect of the shared task was the identification and classification of the maximal NEs as well as the nested NEs, i.e., the constituent parts of larger NEs. But, the training data were provided with the maximal NEs only. For example, *mahatmA gAndhi roDa* (Mahatma Gandhi Road) was annotated as a location and assigned the tag 'NEL' although *mahatmA* (Mahatma) and *gAndhi*(Gandhi) are a person title (NETP) and person name (NEP), respectively. The task was to identify *mahatmA gAndhi roDa* as a NE and classify it as NEL. In addition, *mahatmA*, and *gAndhi* were to be recognized as NEs of the categories NETP (Title), and NEP (Person name). Some NE tags are hard to distinguish in some contexts. For example, it is not always clear whether something should be marked as 'Number' or as 'Measure'. 'Time' and 'Measure' are another confusing pair of NE tags. Another difficult class is 'Technical terms' (NETE) and it is often difficult to decide whether an expression is to be tagged as a 'NETE' or not. For example, it is difficult to decide whether 'Agriculture' is 'NETE', and if not then whether 'Horticulture' is 'NETE' or not. In fact, this is the most difficult class to identify. Other ambiguous tags are 'NETE' and 'NETO' (NE title-objects). We have considered only those NE tags that denote person names, location names, organization names, number expressions, time expressions and measurement expressions. The number, time and measurement expressions are mapped to the *Miscellaneous* tag. Other tags of the shared task have been mapped to the 'other-than-NE' category. Hence, the tag set now becomes as shown in Table 2.

In order to properly denote the boundaries of the NEs, the four NE tags are further subdivided as shown in Table 3. In the output, these sixteen NE tags are directly mapped to the four major NE tags, namely *Person*, *Location*, *Organization* and *Miscellaneous.*

NE Tag	Meaning	Example
NEP	Person name	*sachIna*/NEP, *sachIna ramesha tenDUlkara* / NEP
NEL	Location name	*kolkAtA*/NEL, *mahatmA gAndhi roDa* / NEL
NEO	Organization name	*yadabpUra bishVbidyAlYa*/NEO, *bhAbA eytOmika risArcha sentAra* / NEO
NED	Designation	*cheYArmAn*/NED, *sA.msada*/NED
NEA	Abbreviation	*bi e*/NEA, *ci em di a*/NEA, *bi je pi*/NEA, *Ai.bi.em*/ NEA
NEB	Brand	*fYAntA*/NEB
NETP	Title-person	*shrImAna*/NED, *shrI*/NED, *shrImati*/NED
NETO	Title-object	*AmericAn biUti*/NETO
NEN	Number	*10*/NEN, *dasha*/NEN
NEM	Measure	*tina dina*/NEM, *p.NAch keji*/NEM
NETE	Terms	*hidena markbha madela*/NETE, *kemikYAla riYYAkchYAna*/NETE
NETI	Time	*10 i mAgha 1402* / NETI, *10 ema*/NETI

TABLE 1 Named entity tag set for Indian languages (IJCNLP-08 NERSSEAL Shared Task Tag Set)

IJCNLP-08 shared task tag set	Tag set used	Meaning
NEP	*Person*	Single word/multiword person name
NEL	*Location*	Single word/multiword location name
NEO	*Organization*	Single word/multiword organization name
NEN, NEM, NETI	*Miscellaneous*	Single word/ multiword miscellaneous name
NED, NEA, NEB, NETP, NETE	NNE	Other than NEs

TABLE 2 Tag set used in this work

Named Entity Tag	Meaning	Example
PER	Single word person name	*sachIna*/PER, *rabIndranAtha*/PER
LOC	Single word location name	*kolkAtA*/LOC, *mUmvAi*/LOC
ORG	Single word organization name	*infOsIsa*/ORG
MISC	Single word miscellaneous name	*10*/MISC, *dasha*/MISC
B-PER I-PER E-PER	Beginning, Internal or the End of a multiword person name	*sachIna*/B-PER *ramesha*/I-PER *tenDUlkara* /E-PER, *rabIndranAtha*/B-PER *ThAkUra*/E-PER
B-LOC I-LOC E-LOC	Beginning, Internal or the End of a multiword location name	*mahatmA*/B-LOC *gAndhi* /I-LOC *roDa* /E-LOC, *niU*/B-LOC *iYorka*/E-LOC
B-ORG I-ORG E-ORG	Beginning, Internal or the End of a multiword organization name	*yadabpUra* /B-ORG *bishVbidyAlYa*/E-ORG, *bhAbA* /B-ORG *eytOmika*/I-ORG *risArcha*/I-ORG *sentAra* /E-ORG
B-MISC I-MISC E-MISC	Beginning, Internal or the End of a multiword miscellaneous name	*10 i* /B-MISC *mAgha*/I-MISC *1402*/E-MISC, *10*/B-MISC *ema*/E-MISC
NNE	Other than NEs	*karA*/NNE, *jala*/NNE

TABLE 3 Named entity tag set (B-I-E format)

2.2 A Conditional Random Field Framework for Named Entity Recognition

Indian languages are morphologically very rich and contain non-independent, diverse and overlapping features. A simple Hidden Markov Model (HMM) cannot handle these complex and arbitrary features as efficiently as a Maximum Entropy (ME) (Borthwick 1999), a Conditional Random Field (CRF) (Lafferty et al. 2001) or a Support Vector Machine (SVM) (Yamada et al. 2001) model.

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).

CRF is 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 $o =< o_1, o_2, \ldots, o_T >$ is calculated as:

$$P_\wedge(s|o) = \frac{1}{Z_o} \exp(\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k \times f_k(s_{t-1}, s_t, o, t)),$$

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 $-\infty, \ldots + \infty$, but typically they are binary. To make all conditional probabilities sum up to 1, we must calculate the normalization factor,

$$Z_o = \sum_s \exp(\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k \times f_k(s_{t-1}, s_t, o, t)),$$

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 the observation sequences:

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

where $\{< o^{(i)}, s^{(i)} >\}$ is the labeled training data. The second sum corresponds to a zero-mean, σ^2 -variance Gaussian 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 than Generalized Iterative Scaling or Improved Iterative Scaling, and that results in only minor changes in accuracy due to changes in λ.

When applying CRFs to the NER problem, an observation sequence is a token of a sentence or document of text and the state sequence is its corresponding label sequence. 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 CRF^{++} package [4], a simple, customizable, and open source implementation of CRF for segmenting or labeling sequential data.

3 Named Entity Features for Bengali and Hindi

The templates that define the feature functions play a crucial role in any statistical model. Unlike ME, CRF does not require careful feature selection in order to avoid overfitting. CRF has the freedom to include arbitrary features, and the ability to automatically construct the most useful feature combinations by feature induction. Since, CRFs are log-linear models, and high accuracy may require complex decision boundaries that are non-linear in the space of the original features, the expressive power of the models is often increased by adding new features that are conjunctions of the original features. For example, a conjunction feature might ask if the current word is in the person name list and the next word is a form of an action verb '*ballen*'(told). One could create arbitrary complicated features with these conjunctions. However, it is not feasible to incorporate all possible conjunctions as that might result in a memory overflow.

The main features for the NER task have been identified based on the different possible combinations of the available word and tag context. The features also include prefixes and suffixes for all words. A prefix or suffix is a sequence of the first or last few characters of a word, which may be or not be a linguistically meaningful prefix or suffix. The use of prefix or suffix information works well for highly inflected languages such as the Indian languages. In addition to these, various gazetteer lists have been developed for use in the NER tasks. We have considered different combinations from the following set to find the best set of features for NER in Bengali and Hindi:

F=$\{w_{i-m}, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots w_{i+n}$, |prefix| $\leq n$, |suffix| $\leq n$, NE

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

tag(s) of previous word(s), POS tag(s) of the current and/or the surrounding word(s), First word, Length of the word, Digit information, Infrequent word, Gazetteer lists}, where w_i is the current word; w_{i-m} is the previous mth word and w_{i+n} is the next nth word.

The set 'F' contains both language independent and language dependent features. The set of language independent features includes the context words, the prefixes and suffixes of all the words, dynamic NE information about the previous word(s), first word, length of the word, digit information, infrequent word information and the POS information extracted from language independent POS taggers. Language dependent features for Bengali include the set of known suffixes that may appear with the various NEs, clue words that help predict the location and organization names, words that help recognize measurement expressions, designation words that help to identify person names, various gazetteer lists that include first, middle, and last names, location names, organization names, function words, weekdays and month names. As part of language dependent features for Hindi, the system uses only lists of first, middle, and last names, weekdays, month names along with a list of words that helps recognize measurement expressions. We also include the POS information of the current and/or the surrounding word(s) extracted from the language dependent POS tagger in the set of language dependent features of Bengali.

Language independent NE features can be applied for NER in any language without any prior knowledge of that language. The lists or gazetteers are language dependent at the lexical level but not at the morphological or syntactic level. We include POS information in the set of language independent as well as in the set of language dependent features. The POS information extracted from a language independent POS tagger belongs to the set of language independent features. In addition, several language specific resources such as lexicon, inflection lists and a NER system have been used for another POS tagger with an overall improved performance. This in turn increased the NE tagging accuracy. The POS information extracted from this language dependent POS tagger is regarded as a language dependent feature. The use of language specific features is helpful to improve the performance of the NER system. In the resource-constrained Indian language environment, the need for NER systems acts as a stimulant for the development of currently not always available language specific resources and tools such as POS taggers, gazetteers, morphological analyzers etc. The development of these tools and resources requires knowledge of the language.

3.1 Language Independent Features for Bengali and Hindi

We have considered different combinations of the set of language independent features to select the best set of features for NER in Bengali and Hindi. The following describes the features:

- Context word feature: Words preceding and following a particular word can be used as features. This is based on the observation that the surrounding words are very effective in the identification of NEs.
- Word suffix: Word suffix information is helpful to identify NEs. This is based on the observation that NEs share some common suffixes. This feature can be used in two different ways. The first naïve way to use it is to consider a fixed length (say, n) word suffix of the current and/or the surrounding word(s) as features. This is actually the fixed length character strings (i.e, strings of length 1, 2 0r 3 etc.) stripped from the word endings. If the length of the corresponding word is less than or equal to $n-1$ the feature values are not defined and denoted by ND. The feature value is also not defined (ND) if the token itself is a punctuation symbol or contains a special symbol or digit. The second and more helpful approach is to use the feature as binary valued. Variable length suffixes of a word are matched with predefined lists of useful suffixes for different classes of NEs. Variable length suffixes belong to the category of language dependent features as they require language specific knowledge for their development.
- Word prefix: Word prefixes are also helpful and based on the observation that NEs share common prefix strings. This feature has been defined in a similar way as that of the fixed length suffixes.
- Named Entity Information: The NE tag(s) of the previous word(s) is/are used as the only dynamic feature in the experiment. These tags carry important information in deciding the NE tag of the current word.
- First word: This is used to check whether the current token is the first word of the sentence or not. Though Indian languages are relatively free word order languages, the first word of the sentence is most likely a NE as it is the subject most of the time (i.e., more than 50% cases).
- Digit features: Several binary valued digit features have been defined depending upon the presence and/or the number of digits in a token (e.g., CntDgt [token contains digits], FourDgt [token consists of four digits], TwoDgt [token consists of two digits]), combination of digits and punctuation symbols (e.g., CntDgtCma [token consists of digits and comma], CntDgtPrd [token consists of digits and periods]), combination of digits and symbols (e.g., CntDgtSlsh [token consists

of digit and slash], CntDgtHph [token consists of digits and hyphen], CntDgtPrctg [token consists of digits and percentages]). These binary valued features are helpful to recognize miscellaneous NEs, such as time expressions, measurement expressions and numerical numbers etc.

- Infrequent word: The frequencies of the words in the training corpus have been calculated. A cut off frequency has been chosen. Words that occur less often than the cut off frequency in the training corpus are considered 'infrequent' and listed. A binary valued feature 'Infrequent' is defined to check whether a token appears in this list. The cut off frequencies are set to 10 for Bengali and 20 for Hindi.
- Length of a word: This binary valued feature is used to check whether the length of the current word is less than three or not. This is based on the observation that the very short words are rarely NEs.
- Part of Speech information: POS information has been used as a language independent feature. We have used a CRF-based POS tagger (Ekbal et al. 2007a), which has been developed with a tag set of 27 different POS tags [5], defined for the Indian languages. The POS tagger has been evaluated for Bengali and Hindi without using any language specific resources like lexicons, inflection lists or a NER system. The POS tagger has been trained with the Bengali and Hindi data, obtained through our participations in the NLPAI_Contest06[6] and SPSAL2007[7]competitions. These data sets are different from those used in NER.

The above set of language independent features along with their descriptions are shown in Table 4.

3.2 Language Dependent Features for Bengali and Hindi

Language dependent features for Bengali have been identified based on the earlier experiments (Ekbal and Bandyopadhyay 2007a, Ekbal and Bandyopadhyay 2007c) in NER. Additional NE features have been identified from the Bengali news corpus (Ekbal and Bandyopadhyay 2008b). For Hindi, gazetteers have been prepared manually as well as automatically by processing the data obtained from the Election Commission[8] of India. The various gazetteers used in the experiment are presented in Table 5. These resources will be made available to the public for research use on our personal web pages. Some of the gazetteers that have been used only for Bengali are briefly described below:

[5] http://shiva.iiit.ac.in/SPSAL2007/iiit_tagset_guidelines.pdf

[6] http://ltrc.iiitnet/nlpai_contest06/

[7] http://shiva.iiit.ac.in/SPSAL2007/

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

Feature	Description
ContexT	$ContexT_i = w_{i-m}, \ldots, w_{i-1}, w_i, w_{i+1}, w_{i+n}$, where w_{i-m}, and w_{i+n} are the previous mth, and the next nth word
Suf	$\text{Suf}_i(n) = \begin{cases} \text{Suffix string of length } n \text{ of } w_i \quad \text{if } \lvert w_i \rvert \geq n \\ ND(=0) \;\; \text{if } \lvert w_i \rvert \leq (n-1) \\ \qquad \text{or } w_i \text{ is a punctuation symbol} \\ \qquad \text{or } w_i \text{ contains any special symbol or digit} \end{cases}$
Pre	$\text{Pre}_i(n) = \begin{cases} \text{Prefix string of length } n \text{ of } w_i \quad \text{if } \lvert w_i \rvert \geq n \\ ND(=0) \;\; \text{if } \lvert w_i \rvert \leq (n-1) \\ \qquad \text{or } w_i \text{ is a punctuation symbol} \\ \qquad \text{or } w_i \text{ contains any special symbol or digit} \end{cases}$
NE	$NE_i = \text{NE tag of } w_i$
FirstWord	$\text{FirstWord}_i = \begin{cases} 1, \text{if } w_i \text{ is the first word of a sentence} \\ 0, \text{Otherwise} \end{cases}$
CntDgt	$\text{CntDgt}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit} \\ 0, \text{otherwise} \end{cases}$
FourDgt	$\text{FourDgt}_i = \begin{cases} 1, \text{if } w_i \text{ consists of four digits} \\ 0, \text{otherwise} \end{cases}$
TwoDgt	$\text{TwoDgt}_i = \begin{cases} 1, \text{if } w_i \text{ consists of two digits} \\ 0, \text{otherwise} \end{cases}$
CntDgtCma	$\text{CntDgtCma}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit and comma} \\ 0, \text{otherwise} \end{cases}$
CntDgtPrd	$\text{CntDgtPrd}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit and period} \\ 0, \text{otherwise} \end{cases}$
CntDgtSlsh	$\text{CntDgtSlsh}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit and slash} \\ 0, \text{otherwise} \end{cases}$
CntDgtHph	$\text{CntDgtHph}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit and hyphen} \\ 0, \text{otherwise} \end{cases}$
CntDgtPrctg	$\text{CntDgtPrctg}_i = \begin{cases} 1, \text{if } w_i \text{ contains digit} \\ \qquad \text{and percentage} \\ 0, \text{otherwise} \end{cases}$
Infrequent	$\text{Infrequent}_i = I_{\{\text{Infrequent word list}\}}(w_i)$
Length	$\text{Length}_i = \begin{cases} 1, \text{if } w_i \geq 3 \\ 0, \text{otherwise} \end{cases}$
POS	POS_i=POS tag of the current word

TABLE 4 Descriptions of the language independent features for Bengali and Hindi (Here, i represents the position of the current word and w_i represents the current word)

- NE Suffix list (variable length suffixes): Variable length suffixes of a word are matched with the predefined lists of useful suffixes that are helpful to detect person (e.g., -*bAbU*, -*dA*, -*di* etc.) and location (e.g., -*lyAnDa*, -*pUra*, -*liYA* etc.) names. This has been used only for Bengali.
- Organization suffix word list: This list contains the words that are helpful to identify organization names (e.g., *kO.m*[Co.], *limiteDa*[limited] etc.). These are also part of organization names.
- Person prefix word list: This is useful for detecting person names (e.g., *shrImAna*[Mr.], *shrI*[Mr.], *shrImati*[Mrs.] etc.): person names generally appear after these words.
- Common location word list: This list contains the words (e.g., *saranI*, *rOda*, *lena* etc.) that are part of multiword location names and usually appear at their end.
- Action verb list: A set of action verbs forms like *balena*[told], *balalena*[told], *ballO*[says], *sUnllO*[hears], *h.AsalO*[smiles], *karalO* [did], *gela* [went] etc. often determines the presence of person names. Person names generally appear before the action verbs.
- Designation words: A list of words (e.g., *netA*[leader], *sA.msada*[MP], *khelOYAra*[player] etc.) designating the occupation of persons has been prepared. These words help to identify the position of person names.
- Part of Speech information: Here, the POS information of the words has been regarded as a language dependent features in NE tagging. For POS tagging, we have used a CRF-based POS tagger (Ekbal et al. 2007a), which has been developed with the help of a tag set of 27 different POS tags [9], defined for the Indian languages. We have used inflection lists that can appear with the different word forms of nouns, verbs and adjectives, a lexicon (Ekbal and Bandyopadhyay 2007b) that has been developed in an unsupervised way from the Bengali news corpus, and a CRF based NER system (Ekbal and Bandyopadhyay 2008a) as the features for POS tagging in Bengali. This POS tagger has been developed with the same data set as the language independent POS tagger. This POS tagger yields an accuracy of 90.2%.

The language dependent common features of Bengali and Hindi are represented in Table 6. We have also used a number of features that are unique to Bengali and these are shown in Table 7.

[9] http://shiva.iiit.ac.in/SPSAL2007/iiit_tagset_guidelines.pdf

Language	Gazetteer	Number of entries	Source
Bengali	NE suffix	115	Manually prepared from the news corpus
	Organization suffix	94	Manually created from the news corpus
	Person prefix	245	Manually prepared from the news corpus
	Middle name	1491	Semi-automatically prepared from the news corpus
	Surname	5,288	Semi-automatically prepared from the news corpus
	Common Location	547	Manually prepared from the news corpus
	Action verb	221	Manually prepared from the news corpus
	Designation words	947	Semi-automatically prepared from news corpus
	First names	72,206	Semi-automatically prepared from the news corpus
	Location name	4,875	Semi-automatically prepared from the news corpus
	Organization name	2,225	Manually prepared from the news corpus
	Month name	24	Manually prepared from the news corpus
	Weekdays	14	Manually prepared from the news corpus
	Measurement expressions	52	Manually prepared from the news corpus
Hindi	First name	162,881	Processed from the Election Commission of India data
	Middle name	450	Processed from the Election Commission of India data
	Surname	3,573	Processed from the Election Commission of India data
	Function words	653	Manually prepared
	Month name	24	Manually prepared
	Week days	14	Manually prepared
	Measurement expressions	52	Manually prepared

TABLE 5 Different gazetteers used in the experiment

Feature	Description
FirstName	$FirstName_i = I_{\{\text{First name list}\}}(w_i)$
MidName	$MidName_i = I_{\{\text{Middle name list}\}}(w_i)$
SurName	$SurName_i = I_{\{\text{Sur name list}\}}(w_i) \bigvee I_{\{\text{Sur name list}\}}(w_{i+1})$
Funct	$Funct_i = I_{\{\text{Function word list}\}}(w_i)$
MonthName	$MonthName_i = I_{\{\text{Month name list}\}}(w_i)$
WeekDay	$WeekDay_i = I_{\{\text{Week day list}\}}(w_i)$
MeasureMent	$Measurement_i = I_{\{\text{Measurement word list}\}}(w_{i+1})$ $\bigvee I_{\{\text{Measurement list}\}}(w_{i+1})$

TABLE 6 Descriptions of the language dependent common features for Bengali and Hindi (Here, i represents the position of the current word and w_i represents the current word)

Feature	Description
POS	POS_i=POS tag of the current word
NESuf	$NESuf_i = I_{\{\text{NE suffix list}\}}(w_i)$
OrgSuf	$OrgSuf_i = I_{\{\text{Organization suffix word list}\}}(w_i)$ $\bigvee I_{\{\text{Organization suffix word list}\}}(w_{i+1})$
ComLoc	$ComLoc_i = I_{\{\text{Common location list}\}}(w_i)$
ActVerb	$ActVerb_i = I_{\{\text{Action verb list}\}}(w_i)$ $\bigvee I_{\{\text{Action verb ist}\}}(w_{i+1})$
DesG	$DesG_i = I_{\{\text{Designation word list}\}}(w_{i-1})$
PerPre	$PerPre_i = I_{\{\text{Person prefix word list}\}}(w_{i-1})$
LocName	$LocName_i = I_{\{\text{Location name list}\}}(w_i)$
OrgName	$OrgName_i = I_{\{\text{Organization name list}\}}(w_i)$

TABLE 7 Descriptions of the unique language dependent features for Bengali(Here, i represents the position of the current word and w_i represents the current word)

Set	Bengali	Hindi
Training	102,467 tokens	452,974 tokens
Development	20K tokens	50K tokens
Test	35K tokens	38K tokens

TABLE 8 Statistics of the training, development and test sets

4 Experimental Results

The NER system has been trained with the Bengali and Hindi data, obtained from the IJCNLP-08 NER Shared Task for SSEAL. These twelve NE tagged corpora had to be preprocessed in order to convert them into forms, tagged with the *Person*, *Location*, *Organization* and *Miscellaneous* tags. A subset of each training set has been selected as the development set to identify the best set of features for NER in each of the languages. We use the gold standard test sets to report the evaluation results. Statistics of the training, development and test sets are presented in Table 8.

A feature vector consisting of the features as described in the previous section is extracted for each word in the NE tagged corpus. The training data takes the form (W_i, T_i), where, W_i is the ithword and its feature vector and T_i is its NE tag. Models are built based on training data and the feature template. We have considered various combinations from the set of feature templates as given by,
F_1 =$\{w_{i-m}, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_{i+n}$; Combination of w_{i-1} and w_i; Combination of w_i and w_{i+1}; Feature vector of w_i; POS tags of the current and/or the surrounding word(s); Output tag (t_i) of the previous word; Gazetteer information$\}$

A number of different experiments have been conducted taking the different combinations from the set of features F and the set of feature templates F_1 in order to find the best combination of features and feature templates. Our empirical analysis found that the following combination of features $F_{(best)}$ gives the best result for the development sets. The corresponding feature template is shown in Table 9.

$F_{(best)} = [w_{i-2}w_{i-1}w_iw_{i+1}w_{i+2}, |Pre| \leq 3, |Suf| \leq 3, NE_{i-1}, FirstWord, Length, Infrequent, POS_{i-1}, POS_i, POS_{i+1}, Digitfeatures, Gazetteers]$.

We define the *baseline* model as the one where the NE tag probabilities depend only on the current word:

$$P(t_1, t_2, \ldots, t_n | w_1, w_2, \ldots, w_n) = \prod_{i=1,\ldots,n} P(t_i | w_i)$$

In this model, each word in the test data will be assigned the NE tag

w_{i-2}
w_{i-1}
w_i
w_{i+1}
w_{i+2}
Combination of w_{i-1} and w_i
Combination of w_i and w_{i+1}
Feature vector of w_i
POS tags of w_{i-1}, w_i and w_{i+1}
Output tag (t_i) of the previous word
Gazetteer features

TABLE 9 Feature template for NER in Bengali and Hindi

which is most probable for that word in the training data. Unknown words are assigned the NE tag with the help of various gazetteers and NE suffix lists for Bengali. For Hindi, unknown words are assigned a NE tag on the basis of the limited set of gazetteers and a default NE tag.

4.1 Evaluation of Language Independent Features on the Development Set

Evaluation results for the development sets are presented in Table 10, and Table 11 for Bengali, and Hindi, respectively. Results show (1^{st}-5^{th} rows) that a context word window of size five, i.e., the previous two words, the current word and the next two words, gives the best result (3^{rd} row) along with the 'FirstWord' and 'Length' features. The use of the 'Infrequent' feature increases the f-score values (3^{rd} and 6^{th} rows) by 0.88%, and 0.56% for Bengali, and Hindi, respectively. The NE information of the previous word is the only dynamic feature in the experiment and improves the f-scores by 2.37%, and 2.01% for Bengali, and Hindi, respectively. Usually, NEs contain some common prefixes and suffixes that are very effective for their identification. Evaluation results (8^{th}-15^{th} rows) show a significant improvement in the overall performance of the system in each of the languages due to the effectiveness of prefix and suffix information. The results in row 9^{th} indicate that the prefixes and suffixes of length up to three characters of the current word are more effective than the prefixes and suffixes of higher or lower lengths. Evaluation results (10^{th} -13^{th} rows) also show that surrounding word suffixes and/or prefixes are not as effective as those of the current word. In fact, the inclusion of the surrounding word suffixes and/or prefixes (14^{th} and 15^{th} rows) may degrade the performance of the system. Finally, the use of prefixes and suffixes increases the f-score values of the system to 76.13% (an improvement of 4.71%) for Bengali, and 76.22% (an improvement of 3.18%) for Hindi. We obtain f-score values of 75.21% for Bengali and 77.57% for Hindi with the use of various digit features. Evaluation results show that POS information improves the overall NE tagging accuracies. Results (17^{th}-20^{th} rows) suggest that POS information of the window $[-1, +1]$ is more effective than POS information of the windows $[-1, 0]$, $[0, +1]$ or information for current word alone. We also conducted experiments considering the POS information of windows of [-2, +2], [-2, +1], $[-2, 0]$, $[0, +2]$, $[-2, -1]$ and $[+1, +2]$ and observed lower f-score values for each of the languages. The use of this language independent POS information yields f-score values of 76.85% for Bengali and 78.58% for Hindi. One possible reason behind the better performance for Hindi is the size of the training set, which is approximately five times that of the Bengali training set. Graphical representations of the evaluation results are shown in Figure 1, and Figure 2 for Bengali, and Hindi, respectively.

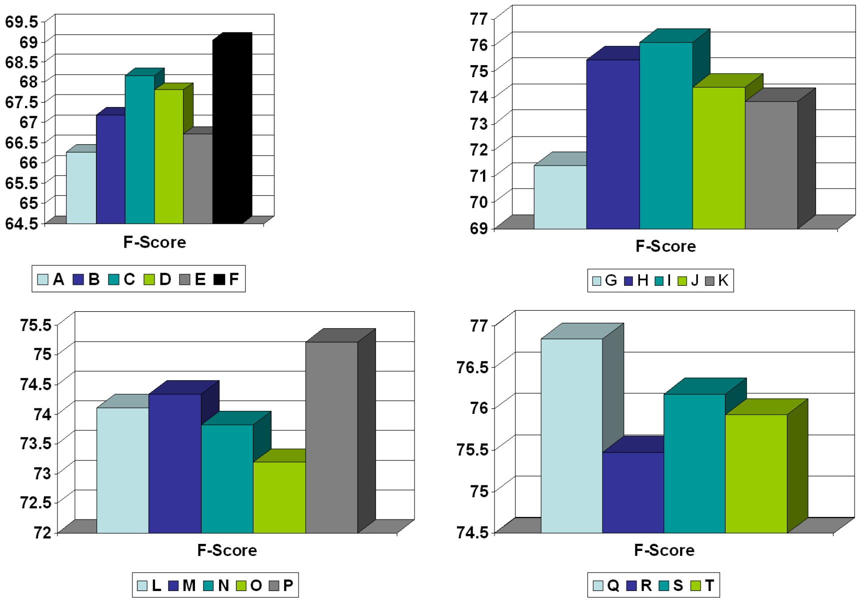

FIGURE 1 Chart of the experimental results on the development set for Bengali using language independent features

Feature (word, tag)	F-Score (in %)
A= w_{i-1}, w_i, w_{i+1}, FirstWord	66.27
B=A+ Length	67.19
C=B+$w_{i-2} + w_{i+2}$	68.17
D=C +w_{i-3}	67.81
E=D +w_{i+3}	66.72
F=C + Infrequent	69.05
G=F++NE_{i-1}	71.42
H=G+$Suf_4(w_i) + Pre_4(w_i)$	75.45
I=G+$Suf_3(w_i) + Pre_3(w_i)$	76.13
J=G+$Suf_3(w_{i-1}) + Pre_3(w_{i-1})$	74.41
K=G+$Suf_3(w_{i+1}) + Pre_3(w_{i+1})$	73.85
L=G+$Pre_3(w_{i+1} + Pre_3(w_{i-1})$	74.11
M=G+$Suf_3(w_{i-1}) + Suf_3(w_{i+1})$	74.34
N=I+$Suf_3(w_{i-1}) + Suf_3(w_{i+1})$	73.82
O=I+$Suf_3(w_{i+1}) + Pre_3(w_{i+1})$	73.19
P=I + Digit features	75.21
Q=P+$POS_{i-1} + POS_i + POS_{i+1}$	76.85
R=P+POS_i	75.48
S=P+$POS_{i-1} + POS_i$	76.18
T=P+$POS_i + POS_{i+1}$	75.93

TABLE 10 Experimental results on the development set for Bengali using language independent features

Feature (word, tag)	F-Score (in %)
A=w_{i-1}, w_i, w_{i+1}, FirstWord	68.15
B=A+ Length	69.24
C=B+$w_{i-2} + w_{i+2}$	70.47
D=C +w_{i-3}	69.82
E=D +w_{i+3}	69.15
F=C + Infrequent	71.03
G=F+NE_{i-1}	73.04
H=G+$Suf_4(w_i) + Pre_4(w_i)$	75.31
I=G+$Suf_3(w_i) + Pre_3(w_i)$	76.22
J=G+$Suf_3(w_{i-1}) + Pre_3(w_{i-1})$	74.94
K=G+$Suf_3(w_{i+1}) + Pre_3(w_{i+1})$	74.08
L=G+$Pre_3(w_{i+1} + Pre_3(w_{i-1})$	74.68
M=G+$Suf_3(w_{i-1}) + Suf_3(w_{i+1})$	75.01
N=I+$Suf_3(w_{i-1}) + Suf_3(w_{i+1})$	74.92
O=I+$Suf_3(w_{i+1}) + Pre_3(w_{i+1})$	74.53
P=I + Digit features	77.57
Q=P+$POS_{i-1} + POS_i + POS_{i+1}$	78.58
R=P+$POS_{i-1} + POS_i$	78.23
S=P+$POS_i + POS_{i+1}$	77.91
$T = P + POS_i$	77.85

TABLE 11 Experimental results on the development set for Hindi using language independent features

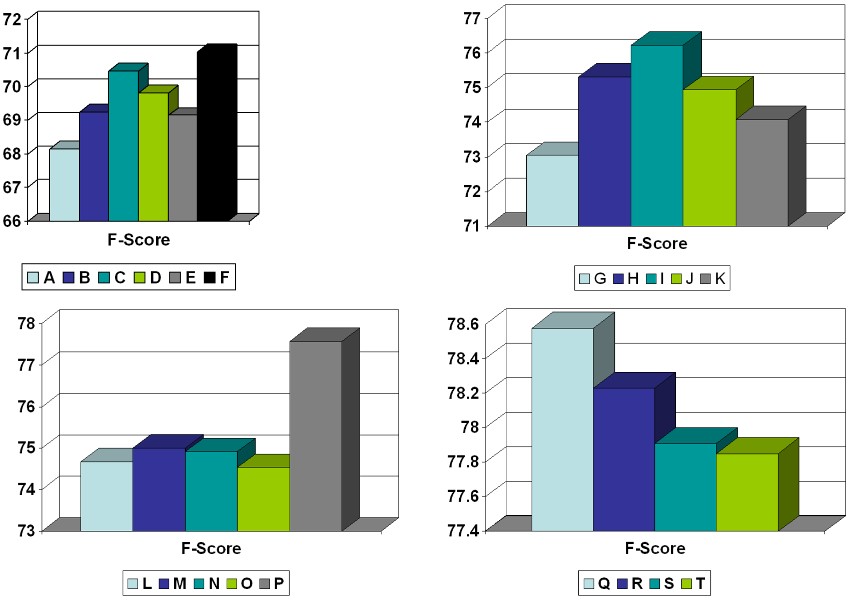

FIGURE 2 Chart of the experimental results on the development set for Hindi using language independent features

4.2 Evaluation of Language Dependent Features on the Development Sets

Evaluation results including the various language dependent features are presented in Table 12, and Table 13 for Bengali, and Hindi, respectively. For Bengali, we have used the POS information extracted from the language dependent POS tagger. This language dependent POS tagger has been developed with the help of different language specific resources such as a lexicon, inflection lists and a NER system. It is evident that language dependent POS tagger increases the f-score value of the NER system by 1.96% (Compare the 17^{th} row of Table 10 and 2^{nd} row of Table 12). Results (2^{nd}-5^{th} rows) suggest that the POS information of the previous, current and the next words yield the best performance with a f-score value of 78.81%.

Next we included a variety of features extracted from the gazetteers in the model. The results of Table 12 (6^{th} row) show that the various suffixes that can occur with the different NEs are very effective in improving the overall performance of the system (an improvement of 1.12%). We also observe (7^{th} row) the effectiveness of the use of orga-

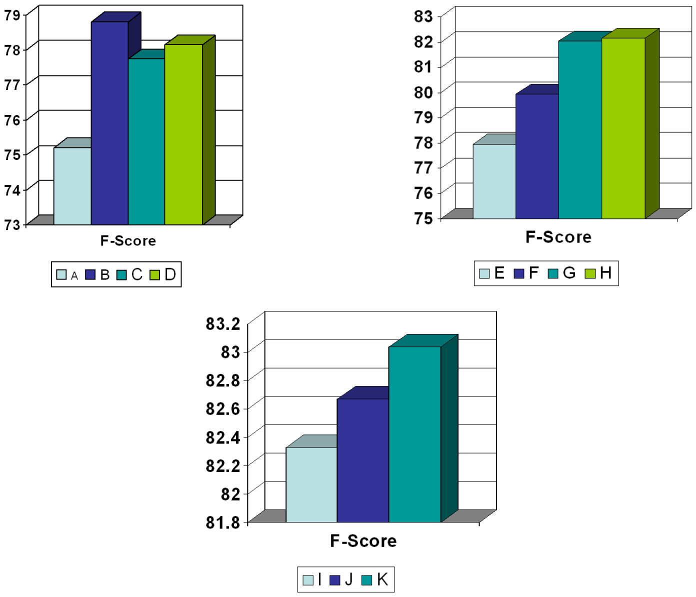

FIGURE 3 Chart of the experimental results on the development set for Bengali by including the language dependent features.

nization suffix words, person prefix words, designations and common location words with an f-score improvement of 2.11%. Other gazetteers improve the performance of the system though their effect are not very impressive. The final system has a f-score value of 83.04% for Bengali. This is improvement of 6.19% (Comparing 17^{th} row of Table 10 and 11^{th} row of Table 12) in f-score value through the use of several language specific features of Bengali. Experimental results of Table 13 show an improvement in f-score by 1.53% with the use several language dependent features for Hindi. The results have been graphically represented in Figure 3, and Figure 4 for Bengali, and Hindi, respectively.

4.3 Evaluation Results of the 10-fold Cross Validation Test

The best set of features for NER in each of the languages is identified by training the CRF based system with 102,467 and 452,974 tokens and testing with the development sets of 20K, and 50K tokens, respectively. We have conducted 10-fold cross validation tests in two different phases, initially with the language independent features (Language independent NER system denoted as LI) and then by including the language dependent features (Language dependent NER system denoted

Feature (word, tag)	F-Score (in %)
A=P of Table 10	75.21
B=A+$POS_{i-1} + POS_i + POS_{i+1}$	78.81
C=A+POS_i	77.75
D=A+$POS_{i-1} + POS_i$	78.16
E=A+$POS_i + POS_{i+1}$	77.95
F=B+ NESuf	79.93
G=F+$OrgSuf + PerPre + DesG + ComLoc$	82.04
H=G+$MidName + SurName + ActVerb$	82.15
I=H+$FirstName + LocName + OrgName$	82.33
J=I+$MonthName + WeekDay$	82.67
K=J+$MeasureMent$	83.04

TABLE 12 Results of the development set for Bengali by including the language dependent features

Feature (word, tag)	F-Score (in %)
A=Q of Table 11	78.58
B=A+$FirstName + MidName + SurName$	79.62
C=B+$MonthName + WeekDay$	79.95
D=C+$MeasureMent$	80.11

TABLE 13 Results of the development set for Hindi by including the language dependent features

Test Set	LI			LD		
No	Recall	Precision	F-Score	Recall	Precision	F-Score
1	79.55	75.27	77.35	85.12	81.97	83.52
2	78.45	74.25	76.29	83.53	80.56	82.02
3	79.57	74.83	77.13	85.54	81.89	83.68
4	79.39	73.45	76.31	86.76	82.15	84.39
5	80.14	75.26	77.62	85.13	82.08	83.58
6	79.56	75.23	77.33	84.86	81.47	83.13
7	80.45	74.76	77.5	85.33	81.69	83.47
8	80.68	76.24	78.39	86.32	81.14	83.68
9	81.34	77.65	79.45	87.78	83.39	85.53
10	81.45	78.68	80.04	87.35	84.57	85.94
Average	80.06	75.56	77.74	85.77	82.09	83.89

TABLE 14 Experimental results of the 10-fold cross validation test for Bengali

Test Set	LI			LD		
No	Recall	Precision	F-Score	Recall	Precision	F-Score
1	78.09	76.35	77.21	82.39	78.09	80.18
2	78.59	76.49	77.53	81.18	77.44	79.27
3	78.62	76.93	77.77	83.06	77.61	80.24
4	77.81	75.32	76.54	84.18	76.07	79.92
5	78.59	75.85	77.19	82.57	77.12	79.75
6	78.92	74.47	76.63	82.53	78.89	80.67
7	76.24	77.99	77.11	84.01	78.03	80.91
8	78.02	76.12	77.06	83.11	81.57	82.33
9	78.12	75.57	76.82	84.17	81.74	82.94
10	78.59	75.29	76.91	84.12	81.98	83.04
Average	78.16	76.04	77.08	83.13	78.85	80.93

TABLE 15 Experimental results of 10-fold cross validation test for Hindi

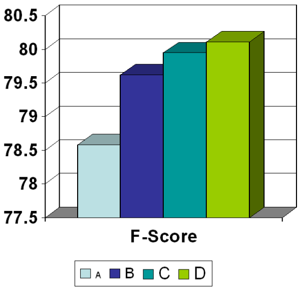

FIGURE 4 Chart of the experimental results on the development set for Hindi using the language dependent features.

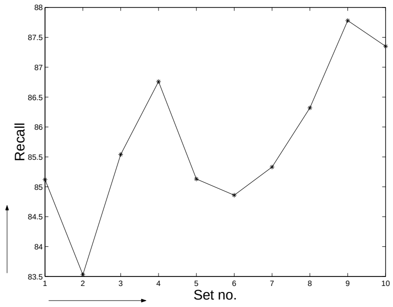

FIGURE 5 Graph showing the recall values of the 10-fold cross validation test for the language dependent Bengali NER system

as LD). Evaluation results of the 10 different experiments in the 10-fold cross validation test are presented in Table 14, and Table 15 for Bengali, and Hindi, respectively. The system shows overall recall, precision, and f-score values of 80.06%, 75.56%, and 77.74%, respectively, for Bengali using the language independent features only. For Hindi, the system shows overall recall, precision, and f-score values of 78.16%, 76.04%, and 77.08%, respectively. The cross validation results also show an improvement in recall, precision, and f-score values by 5.71%, 6.53%, and 6.15%, respectively, for Bengali and 4.97%, 2.81%, and 2.85%, respectively, for Hindi using the language dependent features. The recall, precision, and f-score values using both language independent as well as language dependent features are presented in Figure 5, Figure 6, and Figure 7, for Bengali and in Figure 8, Figure 9, and Figure 10, for Hindi.

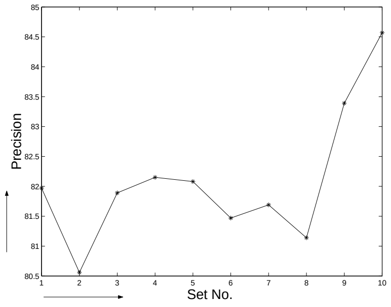

FIGURE 6 Graph showing the precision values of the 10-fold cross validation test for the language dependent Bengali NER system

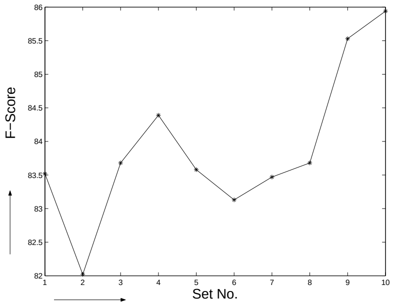

FIGURE 7 Graph showing the f-score values of the 10-fold cross validation test for the language dependent Bengali NER system

In order to show the contribution of the language specific resources such as language dependent POS taggers and gazetteers, we have measured the performance for each of the NE tags. Results are reported in Tables 16-19. Results of Bengali show the highest performance improvement for the *Person* NE tag followed by *Location*, *Organization* and *Miscellaneous* tags. This is due to the use of more linguistic features for person names compared to other NEs. Performance of the other NE tags can be improved by incorporating more linguistic features as like *Person* tag. For Hindi, we also observe the similar nature in the performance as like Bengali.

An ANOVA (Anderson and Scolve 1978) analysis is carried out on the results of 10-fold cross validation test obtained by the language independent CRF based NER system and the language dependent CRF based NER system. Results are reported in Table 20, and Table 21

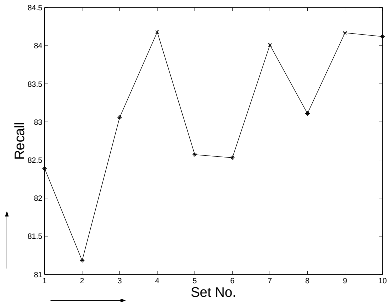

FIGURE 8 Graph showing the recall values of the 10-fold cross validation test for the language dependent Hindi NER system

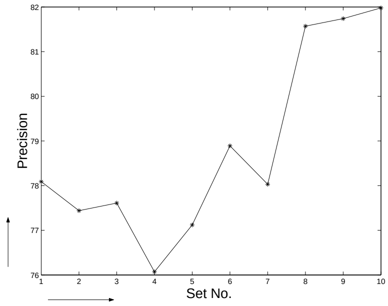

FIGURE 9 Graph showing the precision values for 10-fold cross validation test for the language dependent Hindi NER system

for Bengali, and Hindi, respectively. In the tables, A denotes the language independent NER system and B denotes the language dependent NER system. From the statistical test ANOVA, we can conclude that the difference in the mean recall, precision, and f-score values obtained by the language independent NER system from those obtained by the language dependent NER system for both Bengali and Hindi is statistically significant as in all the cases the significance values are < 0.05.

The box plots (showing the mean and the variances) of the three evaluation criterion for these two approaches are also shown in Figures 11-16.

Tag	Recall	Precision	F-Score
Person	81.23	76.34	78.71
Location	77.69	75.19	76.42
Organization	76.87	74.34	75.58
Miscellaneous	84.12	79.68	81.84

TABLE 16 Results of 10-fold cross validation test for the individual NE tags in the language independent Bengali NER system

Tag	Recall	Precision	F-Score
Person	90.08	88.54	89.3
Location	82.91	78.02	80.39
Organization	78.12	75.21	76.64
Miscellaneous	85.12	80.87	82.94

TABLE 17 Results of the 10-fold cross validation test for the individual NE tags in the language dependent Bengali NER system

Tag	Recall	Precision	F-Score
Person	80.71	76.52	78.56
Location	76.13	73.97	75.03
Organization	75.08	73.01	74.03
Miscellaneous	84.12	80.57	82.31

TABLE 18 Results of the 10-fold cross validation test for the individual NE tags in the language independent Hindi NER system

Tag	Recall	Precision	F-Score
Person	85.71	81.22	83.41
Location	79.43	76.96	78.18
Organization	78.08	76.12	77.09
Miscellaneous	84.23	81.97	83.08

TABLE 19 Results of the 10-fold cross validation test for the individual NE tags in the language dependent Hindi NER system

Evaluation criterion	Tech nique(I)	Mean of I	Comp. Tech.(J)	Mean Diff. (I-J)	Significance value
Recall	A	80.0580 ± 0.8783	B	-5.7140 ± 0.5765	$1.1623e-009$
	B	85.7720 ± 1.6382	A	5.7140 ± 0.5765	$1.1623e-009$
Precision	A	75.5620 ± 2.4654	B	-6.5290 ± 1.0095	$3.3990e-009$
	B	82.0910 ± 1.2985	A	6.5290 ± 1.0095	$3.3990e-009$
F-Score	A	77.7424 ± 1.5085	B	-6.1516 ± 0.5670	$8.6386e-010$
	B	83.89 ± 1.30	A	6.1516 ± 0.5670	$8.6386e-010$

TABLE 20 Estimated marginal means and pairwise comparison of the language independent Bengali NER system (A) and the language dependent Bengali NER system (B)

Evaluation criterion	Tech nique(I)	Mean of I	Comp. Tech.(J)	Mean Diff. (I-J)	Significance value
Recall	A	78.1590 ± 0.5754	B	-4.9730 ± 2.2798	$2.4656e-010$
	B	83.132 ± 0.9964	A	4.9730 ± 2.2798	$2.4656e-010$
Precision	A	76.0380 ± 0.9669	B	-2.8160 ± 6.5928	0.0013
	B	78.8540 ± 4.5606	A	2.8160 ± 6.5928	0.0013
F-Score	A	77.0765 ± 0.1428	B	-3.8485 ± 2.3804	$8.5794e-008$
	B	80.9250 ± 1.8585	A	3.8485 ± 2.3804	$8.5794e-008$

TABLE 21 Estimated marginal means and pairwise comparison of the language independent Hindi NER system (A) and the language dependent Hindi NER system (B)

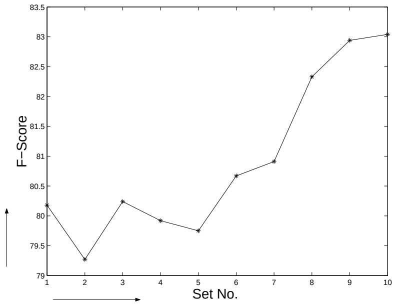

FIGURE 10 Graph showing the f-score values of the 10-fold cross validation test the language dependent Hindi NER system

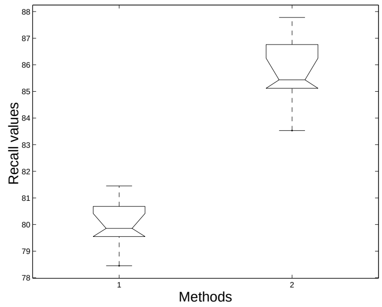

FIGURE 11 Box plot of the recall values obtained by the language independent Bengali NER system (1) and the language dependent Bengali NER system (2)

4.4 Evaluation Results of the Test Set

We have used two gold standard test sets to report the evaluation results of the system for Bengali and Hindi. For Bengali, this test set has been manually prepared by annotating a portion of the Bengali news corpus (Ekbal and Bandyopadhyay 2008b). We have used the gold standard test set obtained from the IJCNLP-08 NERSSEAL shared task for Hindi. This shared task test set has been converted to take only the maximal NEs into consideration. Initially, the system has been evaluated using only the language independent features. Then, the models have been retrained by including the language dependent features. Evaluation results of the system along with the *baseline* models are presented in Table 22 for both of the languages. Results show the performance improvement for both languages with the use of language

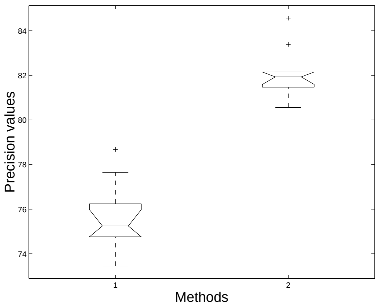

FIGURE 12 Box plot of the precision values obtained by the language independent Bengali NER system (1) and the language dependent Bengali NER system (2)

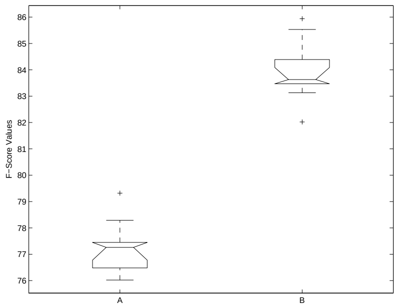

FIGURE 13 Box plot of the f-score values obtained by the language independent Bengali NER system (1) and the language dependent Bengali NER system (2)

dependent features. The higher rate of performance improvement for Bengali is due to the use of more language specific features compared to Hindi. We have also evaluated the systems for each of the individual NE tags. The results are presented in Tables 23-26. Results show the effectiveness of the language specific resources through the improvement in performance in both languages.

In order to improve the system performance, we need to analyze and understand where it went wrong. We have conducted error analysis with a confusion matrix, also called a contingency table. The confusion matrices have been shown in Tables 27-30 for both languages.

Confusion matrices show that the most confusing pairs of classes are *Person* vs NNE, *Location* vs NNE, *Organization* vs NNE, *Person* vs *Organization*, and *Location* vs *Organization* in both languages. The

Model	Recall	Precision	F-Score
Baseline-B	61.08	53.97	57.31
CRF-B (language independent)	76.49	75.09	75.78
CRF-B (language dependent)	82.71	79.65	81.15
Baseline-H	62.56	51.22	56.32
CRF-H (language independent)	77.34	75.93	76.63
CRF-H (language dependent)	80.54	76.16	78.29

TABLE 22 Experimental results of the test set (CRF-B: Model used for Bengali, CRF-H: Model used for Hindi)

Tag	Recall	Precision	F-Score
Person	77.35	75.19	76.25
Location	75.59	73.28	74.42
Organization	74.02	71.93	72.96
Miscellaneous	81.17	76.11	78.56

TABLE 23 Results on the test set for the individual NE tag in the language independent Bengali NER system

Tag	Recall	Precision	F-Score
Person	89.78	84.23	86.92
Location	83.91	81.08	82.47
Organization	78.23	76.01	77.10
Miscellaneous	82.69	77.87	80.21

TABLE 24 Results on the test set for the individual NE tag in the language dependent Bengali NER system

Tag	Recall	Precision	F-Score
Person	79.12	77.17	78.13
Location	77.09	75.11	76.09
Organization	75.22	72.89	74.04
Miscellaneous	81.69	77.83	79.71

TABLE 25 Results on the test set for the individual NE tag in the language independent Hindi NER system

Tag	Recall	Precision	F-Score
Person	85.17	78.86	81.89
Location	77.51	75.13	76.3
Organization	75.97	74.03	74.99
Miscellaneous	81.94	77.67	79.75

table 26 Results on the test set for the individual NE tag in the language dependent Hindi NER system

	Person	*Location*	*Organization*	*Miscellaneous*	NNE
Person	–	2.101	3.302	1.001	9.103
Location	1.519	–	2.607	1.009	10.108
Organization	4.851	6.131	–	1.002	11.336
Miscellaneous	0.61	0.63	0.64	–	6.12
NNE	0.14	0.19	0.17	0.11	–

table 27 Confusion matrix of the language independent Bengali NER system

	Person	*Location*	*Organization*	*Miscellaneous*	NNE
Person	-	.835	1.534	0.72	4.227
Location	.021	-	0.82	0.67	6.055
Organization	1.522	2.083	-	1.034	7.856
Miscellaneous	0.51	0.56	0.61	-	4.34
NNE	0.12	0.14	0.15	0.09	-

table 28 Confusion matrix of the language dependent Bengali NER system

	Person	*Location*	*Organization*	*Miscellaneous*	NNE
Person	-	1.123	2.001	1.11	8.122
Location	2.101	-	3.212	1.111	9.563
Organization	3.554	4.102	-	2.22	8.112
Miscellaneous	0.79	0.72	0.49	-	7.423
NNE	0.39	0.26	0.432	0.22	-

table 29 Confusion matrix of the language independent Hindi NER system

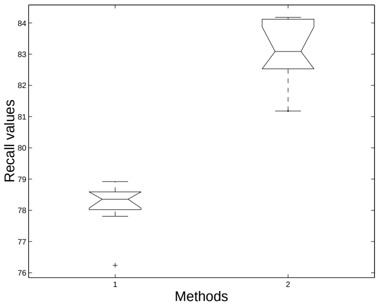

FIGURE 14 Box plot of the recall values obtained by the language independent Hindi NER system (1) and the language dependent Hindi NER system (2)

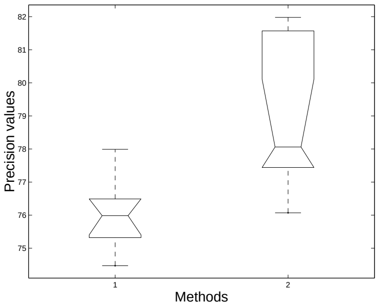

FIGURE 15 Box plot of the precision values obtained by the language independent Hindi NER system (1) and the language dependent Hindi NER system (2)

current system assigns tags to each sentence of the test set by considering the best tag only. The first three errors can be reduced by considering the n-best results for every sentence of the test set. Confusion matrices of Tables 27-30 show that the errors can be reduced considerably by using the language dependent features. Errors involving *Person* vs *Organization*, and *Location* vs *Organization* can be further reduced by post-processing the output of the CRF model with the various gazetteers (e.g., Organization suffix word list, person, location, organization etc.).

4.5 Comparison with the Other NER Systems

We trained and tested the other existing Bengali NER systems (Ekbal and Bandyopadhyay 2007a, Ekbal et al. 2007b) under the same ex-

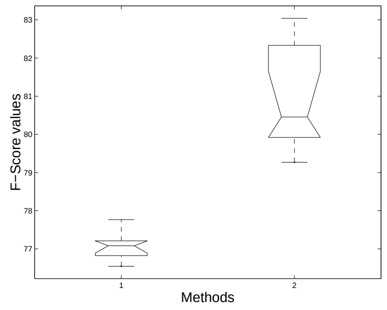

FIGURE 16 Box plot of the f-score values obtained by the language independent Hindi NER system (1) and the language dependent Hindi NER system (2)

	Person	*Location*	*Organization*	*Miscellaneous*	NNE
Person	-	.07	1.104	1.57	7.78
Location	1.709	-	2.132	0.98	8.131
Organization	2.177	3.098	-	1.83	6.899
Miscellaneous	0.21	0.45	0.51	-	6.342
NNE	0.32	0.122	0.33	0.19	-

TABLE 30 Confusion matrix of the language dependent Hindi NER system

perimental setup. Two models, namely A and B, have been reported in Ekbal and Bandyopadhyay (2007a). These models are based on a pattern directed shallow parsing approach. An unsupervised algorithm was developed that tags the unlabeled corpus with the seed entities of *Person*, *Location* and *Organization*. Model A uses only the seed lists to tag the training corpus whereas in model B, we have used the various gazetteers along with the seed entities for tagging. The lexical context patterns generated in this way are used to generate further patterns by bootstrapping. The algorithm terminates when no new patterns can be generated. During testing, model A could not deal with the *NE classification disambiguation* problem (i.e., it can not handle the situation when a particular word is tagged with more than one NE type) but model B can handle this problem with the help of gazetteers and various language dependent features.

A HMM-based NER system has been described in Ekbal et al. (2007b), where more context information has been taken into consideration during emission probabilities and where word suffixes have been used to handle the unknown words. We have post-processed the output of the HMM-based system with the lexical context patterns generated from model B. Comparative evaluation results for the test set are presented in Table 31. Results show that the proposed system outperforms the least performing model A by 12.83% and the best performing HMM-based system by 5.25%. One reason for this increase in recall, precision and f-score values in the CRF based NER system is its ability to handle the non-independent, diverse and overlapping features of the morphologically rich Indian languages more efficiently than the HMM.

Comparisons with the studies described in the IJCNLP-08 shared task are not possible for the following reasons:

- The shared task used a fine-grained tag set of twelve NE tags. In this work, we have considered only the tags that denote person name, location name, organization name, date, time and number expressions.
- The main challenge of the shared task was to identify and classify the nested NEs (i.e., the constituent parts of a bigger NE). Here, we are not concerned with the nested NEs.

A CRF-based Hindi NER system with an automatic feature induction technique has been described in Li and McCallum (2004). They trained their system with 340K word forms and tested it with 10K word forms. They literally guessed the relevant features and discovered the useful ones with a feature induction method. They have used atomic features that include the entire word text, character n-grams (n=2,3,4), word prefix and suffix of lengths 2, 3, and 4, and 24 Hindi gazetteers that were provided at the Surprise Language resource website. The feature induction procedure (McCallum 2003) made these atomic features available for the current, previous, and next sequence. They experimented with various options, such as first-order versus second-order models, using lexicons and using feature induction. They also tried different Gaussian priors and early stopping in order to avoid overfitting. Their best performance is an f-score of 82.55% for the 10-fold cross validation test in a first-order model with a early stopping point of 240 iterations of L-BFGS.

In this work, we studied the best combination of features for each of the languages using a development set. We conducted a number of experiments making various combinations of features and feature templates. A particular feature is only included into the set of features if the resultant model performs better than the preceding model. A feature combination is not considered if it degrades the performance. The feature induction method reported in this paper is analogous to that of McCallum (2003). This method is founded on the principle of iteratively constructing feature conjunctions that would significantly

Model	Recall	Precision	F-Score
A	69.57	67.12	68.32
B	72.17	70.09	71.11
HMM	77.73	74.15	75.9
CRF (language dependent)	82.71	79.65	81.15

TABLE 31 Comparative evaluation results for Bengali (A: Pattern directed shallow parsing approach without linguistic knowledge, B: Pattern directed shallow parsing approach with linguistic knowledge)

increase the conditional log-likelihood if added to the model. It has been reported that this feature induction enables not only improved accuracy and dramatic reduction in parameter count, but also the use of larger cliques, and more freedom to liberally hypothesize atomic input variables that may relevant to a task. We experimented with the same set of features with this feature induction technique and observed an improvement in the f-score value of 1.3% compared to the proposed system for Hindi.

5 Conclusion

In this paper, we presented the NER systems for two leading Indian languages (ILs), Bengali and Hindi using CRF. The system makes use of different types of contextual information along with different orthographic word-level features that are helpful in predicting four different NE classes. We have used the IJCNLP-08 NERSSEAL shared task data that was tagged with a fine-grained tag set of twelve tags. We have considered only those tags that denote person, location, organization and miscellaneous (time, measurement and number expressions) names. The system uses language independent features that are applicable to both languages as well as language specific features of Bengali and Hindi. The system obtains f-score values of 81.15%, and 78.29% for Bengali and Hindi, respectively. We have also shown that the use of language dependent resources (or, features) can improve the performance of the system. It has been also shown that the proposed CRF based system outperforms three other existing Bengali NER systems. An ANOVA statistical analysis has been performed to show that the performance improvement of the language dependent NER system over the language independent NER system is statistically significant for both languages.

The performance of the system can further be improved for Hindi by developing other gazetteers and using other language specific features as was done for Bengali. A detailed look at the evaluation results show that the system performs poorly in some specific cases such as for those words that are NEs but that also appear in a common noun dictionary. Some contextual information may be helpful to resolve such ambiguities.

References

Anderson, T. W. and S.L. Scolve. 1978. *Introduction to the Statistical Analysis of Data*. Houghton Mifflin.

Aone, Chinatsu, L. Halverson, T. Hampton, and M. Ramos-Santacruz. 1998. SRA: Description of the IE2 system used for MUC-7. In *MUC-7*. Fairfax, Virginia.

Babych, Bogdan and A. Hartley. 2003. Improving Machine Translation Quality with Automatic Named Entity Recognition. In *Proceedings of EAMT/EACL 2003 Workshop on MT and other Language Technology Tools*, pages 1–8.

Bennet, Scott W., C. Aone, and C. Lovell. 1997. Learning to Tag Multilingual Texts Through Observation. In *Proceedings of Empirical Methods of Natural Language Processing*, pages 109–116. Providence, Rhode Island.

Bikel, Daniel M., Richard L. Schwartz, and Ralph M. Weischedel. 1999. An Algorithm that Learns What's in a Name. *Machine Learning* 34(1-3):211–231.

Borthwick, A. 1999. *Maximum Entropy Approach to Named Entity Recognition*. Ph.D. thesis, New York University.

Borthwick, Andrew, J. Sterling, E. Agichtein, and R. Grishman. 1998. NYU:Description of the MENE Named Entity System as Used in MUC-7. In *MUC-7*. Fairfax.

Burger, John D., John C. Henderson, and T. Morgan. 2002. Statistical Named Entity Recognizer Adaption. In *Proceedings of the CoNLL Workshop*, pages 163–166. Taipei, Taiwan.

Carrears, Xavier, Liuis Marquez, and Liuis Padro. 2002. Named Entity Recognition using AdaBoost. In *Proceedings of the CoNLL Workshop*, pages 167–170. Taipei, Taiwan.

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

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

Cucerzon, S. and David Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the 1999 Joint SIGDAT conference on EMNLP and VLC*. Washington, D.C.

Cunningham, H. 2002. GATE, a General Architecture for Text Engineering. *Computers and the Humanities* 36:223–254.

Ekbal, A. and S. Bandyopadhyay. 2007a. Lexical Pattern Learning from Corpus Data for Named Entity Recognition. In *Proceedings of ICON*, pages 123–128. India.

Ekbal, A. and S. Bandyopadhyay. 2007b. Lexicon Development and POS Tagging using a Tagged Bengali News Corpus. In *Proceedings of the 20th International Florida AI Research Society Conference (FLAIRS-2007)*, pages 261–263. Florida.

Ekbal, A. and S. Bandyopadhyay. 2007c. Pattern Based Bootstrapping Method for Named Entity Recognition. In *Proceedings of ICAPR*, pages 349–355. India.

Ekbal, A. and S. Bandyopadhyay. 2008a. Bengali Named Entity Recognition using Support Vector Machine. In *Proceedings of Workshop on NER for South and South East Asian Languages, 3rd International Joint Conference on Natural Languge Processing (IJCNLP)*, pages 51–58. India.

Ekbal, A. and S. Bandyopadhyay. 2008b. A Web-based Bengali News Corpus for Named Entity Recognition. *Language Resources and Evaluation Journal* 42(2):173–182.

Ekbal, Asif, Rejwanul Haque, and Sivaji Bandyopadhyay. 2007a. Bengali Part of Speech Tagging using Conditional Random Field. In *Proceedings of Seventh International Symposium on Natural Language Processing (SNLP2007)*, pages 131–136. Thailand.

Ekbal, A., S.K. Naskar, and S. Bandyopadhyay. 2007b. Named Entity Recognition and Transliteration in Bengali. *Named Entities: Recognition, Classification and Use, Special Issue of Lingvisticae Investigationes Journal* 30(1):95–114.

Ekbal, A., R.Haque, and S. Bandyopadhyay. 2008. Named Entity Recognition in Bengali: A Conditional Random Field Approach . In *Proceedings of the 3rd International Joint Conference on Natural Language Processing (IJCNLP 2008)*, pages 589–594.

Gali, Karthik, Harshit Sharma, Ashwini Vaidya, Praneeth Shisthla, and Dipti Misra Sharma. 2008. Aggregrating Machine Learning and Rule-based Heuristics for Named Entity Recognition. In *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 25–32.

Humphreys, K., R. Gaizauskas, S. Azzam, C. Huyck, B. Mitchell, H. Cunnigham, and Y. Wilks. 1998. Univ. Of Sheffield: Description of the LaSIE-II System as Used for MUC-7. In *MUC-7*. Fairfax, Virginia.

Kumar, N. and Pushpak Bhattacharyya. 2006. Named entity recognition in hindi using memm. Technical report, IIT Bombay, India.

Kumar, P. Praveen and V. Ravi Kiran. 2008. A Hybrid Named Entity Recognition System for South Asian Languages. In *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 83–88.

Lafferty, John D., Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*, pages 282–289.

Li, Wei and Andrew McCallum. 2004. Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction. *ACM Transactions on Asian Languages Information Processing* 2(3):290–294.

Malouf, Robert. 2002. Markov Models for Language Independent Named Entity Recognition. In *Proceedings of the CoNLL Workshop*, pages 187–190. Taipei, Taiwan.

McCallum, Andrew. 2003. Efficiently Inducing Features of Conditional Random Fields. In *Proceedings of the 19th Conference on Uncertaininty in Artificial Intelligence (UAI03).*

McCallum, A. and W. Li. 2003. Early results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-enhanced Lexicons. In *Proceedings of CoNLL*, pages 188–191. Canada.

Mikheev, A., C. Grover, and M. Moens. 1998. Description of the LTG system used for MUC-7. In *MUC-7*. Fairfax, Virginia.

Mikheev, A., C. Grover, and M. Moens. 1999. Named Entity Recognition without Gazeteers. In *Proceedings of EACL*, pages 1–8. Bergen, Norway.

Miller, S., M. Crystal, H. Fox, L. Ramshaw, R. Schawartz, R. Stone, R. Weischedel, and the Annotation Group. 1998. BBN: Description of the SIFT System as Used for MUC-7. In *MUC-7*. Fairfax, Virginia.

Moldovan, D., S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan. 2002. LCC Tools for Question Answering. In *Text REtrieval Conference (TREC) 2002*.

Pinto, D., A. McCallum, X. Wei, , and W. B. Croft. 2003. Table Extraction using Conditional Random Fields. In *Proceedings of SIGIR'03 Conference*.

Saha, S., S. Sarkar, and P. Mitra. 2008. A Hybrid Feature set based Maximum Entropy Hindi Named Entiy Recognition. In *Proceedings of the 3rd International Joint Conference in Natural Langauge Processing (IJCNLP 2008)*, pages 343–350.

Sekine, Satoshi. 1998. Description of the japanese ne system used for met-2. In *MUC-7*. Fairfax, Virginia.

Sha, Fei and Fernando Pereira. 2003. Shallow Parsing with Conditional Random Fields. In *Proceedings of NAACL '03*, pages 134–141. Canada.

Shishtla, Praneeth M, Prasad Pingali, and Vasudeva Varma. 2008. A Character n-gram Based Approach for Improved Recall in Indian Language ner. In *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 101–108.

Srikanth, P and Kavi Narayana Murthy. 2008. Named Entity Recognition for Telugu. In *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 41–50.

Vijayakrishna, R. and L. Sobha. 2008. Domain Focused Named Entity Recognizer for Tamil using Conditional Random Fields. In *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 93–100.

Yamada, Hiroyasu, Taku Kudo, and Yuji Matsumoto. 2001. Japanese Named Entity Extraction using Support Vector Machine. *In Transactions of IPSJ* 43(1):44–53.

Zhou, GuoDong and Jian Su. 2002. Named Entity Recognition using an HMM-based Chunk Tagger. In *Proceedings of ACL*, pages 473–480. Philadelphia.