Bengali Named Entity Recognition using Support Vector Machine

Asif Ekbal
Department of Computer Science and Engineering, Jadavpur University
Kolkata-700032, India
asif.ekbal@gmail.com

Sivaji Bandyopadhyay
Department of Computer Science and Engineering, Jadavpur University
Kolkata-700032, India
svaji_cse_ju@yahoo.com

Abstract

Named Entity Recognition (NER) aims to classify each word of a document into predefined target named entity classes and is nowadays considered to be fundamental for many Natural Language Processing (NLP) tasks such as information retrieval, machine translation, information extraction, question answering systems and others. This paper reports about the development of a NER system for Bengali using Support Vector Machine (SVM). Though this state of the art machine learning method has been widely applied to NER in several well-studied languages, this is our first attempt to use this method to Indian languages (ILs) and particularly for Bengali. 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. A portion of a partially NE tagged Bengali news corpus, developed from the archive of a leading Bengali newspaper available in the web, has been used to develop the SVM-based NER system. The training set consists of approximately 150K words and has been manually annotated with the sixteen NE tags. Experimental results of the 10-fold cross validation test show the effectiveness of the proposed SVM based NER system with the overall average Recall, Precision and F-Score of 94.3%, 89.4% and 91.8%, respectively. It has been shown that this system outperforms other existing Bengali NER systems.

1 Introduction

Named Entity Recognition (NER) is an important tool in almost all NLP application areas such as information retrieval, machine translation, question-answering system, automatic summarization etc. Proper identification and classification of NEs are very crucial and pose a very big challenge to the NLP researchers. The level of ambiguity in NER makes it difficult to attain human performance

NER has drawn more and more attention from the NE tasks (Chinchor 95; Chinchor 98) in Message Understanding Conferences (MUCs) [MUC6; MUC7]. The problem of correct identification of NEs 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 Hidden Markov Model (HMM) (BBN's IdentiFinder in (Bikel, 1999)), Maximum Entropy (New York University's MEME in (Borthwick, 1999)), Decision Tree (New York University's system in (Sekine, 1998) and Conditional Random Fields (CRFs) (Lafferty et al., 2001). Support Vector Machines (SVMs) based NER system was proposed by Yamada et al. (2002) for Japanese. His system is an extension of Kudo's chunking system (Kudo and Matsumoto, 2001) that gave the best performance at CoNLL-2000 shared tasks. The other SVM-based NER systems can be found in (Takeuchi and Collier, 2002) and (Asahara and Matsumoto, 2003).

Named entity identification in Indian languages in general and particularly in Bengali is difficult and challenging. In English, the NE always appears with capitalized letter but there is no concept of capitalization in Bengali. There has been a very

Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 51–58,
Hyderabad, India, January 2008.

little work in the area of NER in Indian languages. In Indian languages, particularly in Bengali, the works in NER can be found in (Ekbal and Bandyopadhyay, 2007a; Ekbal and Bandyopadhyay, 2007b) with the pattern directed shallow parsing approach and in (Ekbal et al., 2007c) with the HMM. Other than Bengali, a CRF-based Hindi NER system can be found in (Li and McCallum, 2004).

The rest of the paper is organized as follows. Support Vector Machine framework is described briefly in Section 2. Section 3 deals with the named entity recognition in Bengali that describes the named entity tagset and the detailed descriptions of the features for NER. Experimental results are presented in Section 4. Finally, Section 5 concludes the paper.

2 Support Vector Machines

Support Vector Machines (SVMs) are relatively new machine learning approaches for solving two-class pattern recognition problems. SVMs are well known for their good generalization performance, and have been applied to many pattern recognition problems. In the field of NLP, SVMs are applied to text categorization, and are reported to have achieved high accuracy without falling into over-fitting even though with a large number of words taken as the features.

Suppose we have a set of training data for a two-class problem: $\{(x_1, y_1), ...(x_N, y_N)\}$, where $x_i \in R^D$ is a feature vector of the i-th sample in the training data and $y_i \in \{+1, -1\}$ is the class to which x_i belongs. The goal is to find a decision function that accurately predicts class y for an input vector x. A non-linear SVM classifier gives a decision function $f(x) = sign(g(x))$ for an input vector where,

$$g(x) = \sum_{i=1}^{m} w_i K(x, z_i) + b$$

Here, $f(x) = +1$ means x is a member of a certain class and $f(x) = -1$ means x is not a member. z_i s are called support vectors and are representatives of training examples, m is the number of support vectors. Therefore, the computational complexity of $g(x)$ is proportional to m. Support vectors and other constants are determined by solving a certain quadratic programming problem. $K(x, z_i)$ is a *kernel* that implicitly maps vectors into a higher dimensional space. Typical kernels use dot products: $K(x, z_i) = k(x.z)$. A polynomial kernel of degree d is given by $K(x, z_i) = (1 + x)^d$. We can use various kernels, and the design of an appropriate kernel for a particular application is an important research issue.

We have developed our system using SVM (Jochims, 1999) and (Valdimir, 1995), which performs classification by constructing an N-dimensional hyperplane that optimally separates data into two categories. Our general NER system includes two main phases: training and classification. Both the training and classification processes were carried out by *YamCha*[1] toolkit, an SVM based tool for detecting classes in documents and formulating the NER task as a sequential labeling problem. Here, the pair wise multi-class decision method and *second degree polynomial kernel function* were used. We have used TinySVM-0.07[2] classifier that seems to be the best optimized among publicly available SVM toolkits.

3 Named Entity Recognition in Bengali

Bengali is one of the widely used languages all over the world. It is the seventh popular language in the world, second in India and the national language of Bangladesh. A partially NE tagged Bengali news corpus (Ekbal and Bandyopadhyay, 2007d), developed from the archive of a widely read Bengali newspaper. The corpus contains around 34 million word forms in ISCII (Indian Script Code for Information Interchange) and UTF-8 format. The *location, reporter, agency* and different *date* tags (*date, ed, bd, day*) in the partially NE tagged corpus help to identify some of the location, person, organization and miscellaneous names, respectively that appear in some fixed places of the newspaper. These tags cannot detect the NEs within the actual news body. The date information obtained from the news corpus provides example of miscellaneous names. A portion of this partially NE tagged corpus has been manually annotated with the sixteen NE tags as described in Table 1.

3.1 Named Entity Tagset

A SVM based NER system has been developed in this work to identify NEs in Bengali and classify

[1]http://chasen-org/~taku/software/yamcha/
[2]http://cl.aist-nara.ac.jp/~taku-ku/software/TinySVM

them into the predefined four major categories, namely, 'Person name', 'Location name', 'Organization name' and 'Miscellaneous name'. In order to properly denote the boundaries of the NEs and to apply SVM in NER task, sixteen NE and one non-NE tags have been defined as shown in Table 1. In the output, sixteen NE tags are replaced appropriately with the four major NE tags by some simple heuristics.

NE tag	Meaning	Example
PER	Single word person name	*sachin* / PER
LOC	Single word location name	*jdavpur*/LOC
ORG	Single word organization name	*infosys* / ORG
MISC	Single word miscellaneous name	100%/ MISC
B-PER I-PER E-PER	Beginning, Internal or the End of a multiword person name	*sachin*/B-PER *ramesh*/I-PER *tendulkar*/E-PER
B-LOC I-LOC E-LOC	Beginning, Internal or the End of a multiword location name	*mahatma*/B-LOC *gandhi*/I-LOC *road*/E-LOC
B-ORG I-ORG E-ORG	Beginning, Internal or the End of a multiword organization name	*bhaba*/B-ORG *atomic*/I-ORG *research*/I-ORG *center*/E-ORG
B-MISC I-MISC E-MISC	Beginning, Internal or the End of a multiword miscellaneous name	*10e*/B-MISC *magh*/I-MISC *1402*/E-MISC
NNE	Words that are not named entities	*neta*/NNE, *bidhansabha*/NNE

Table 1. Named Entity Tagset

3.2 Named Entity Feature Descriptions

Feature selection plays a crucial role in the Support Vector Machine (SVM) framework. Experiments have been carried out in order to find out the most suitable features for NER in Bengali. 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 for use in the NER task. We have considered different combination from the following set for inspecting the best feature set for NER task:

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

Following are the details of the set of features that have been applied to the NER task:

- Context word feature: Previous and next words of a particular word might be used as a feature.
- Word suffix: Word suffix information is helpful to identify NEs. This feature can be used in two different ways. The first and the naïve one is, a fixed length word suffix of the current and/or the surrounding word(s) might be treated as feature. The second and the more helpful approach is to modify the feature as binary valued. Variable length suffixes of a word can be matched with predefined lists of useful suffixes for different classes of NEs. The different suffixes that may be particularly helpful in detecting person (e.g., *-babu*, *-da*, *-di* etc.) and location names (e.g., *-land*, *-pur*, *-lia* etc.) are also included in the lists of variable length suffixes. Here, both types of suffixes have been used.
- Word prefix: Prefix information of a word is also helpful. A fixed length prefix of the current and/or the surrounding word(s) might be treated as features.
- Part of Speech (POS) Information: The POS of the current and/or the surrounding word(s) can be used as features. Multiple POS information of the words can be a feature but it has not been used in the present work. The alternative and the better way is to use a coarse-grained POS tagger.

Here, we have used a CRF-based POS tagger, which was originally developed with the help of 26 different POS tags[3], defined for Indian languages. For NER, we have considered a coarse-grained POS tagger that has only the following POS tags:

NNC (Compound common noun), NN (Common noun), NNPC (Compound proper noun), NNP (Proper noun), PREP (Postpositions), QFNUM (Number quantifier) and Other (Other than the above).

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

The POS tagger is further modified with two POS tags (Nominal and Other) for incorporating the nominal POS information. Now, a binary valued feature 'nominalPOS' is defined as: If the current/surrounding word is 'Nominal' then the 'nominalPOS' feature of the corresponding word is set to '+1'; otherwise, it is set to '-1'. This binary valued 'nominalPOS' feature has been used in addition to the 7-tag POS feature. Sometimes, postpositions play an important role in NER as postpositions occur very frequently after a NE. A binary valued feature 'nominalPREP' is defined as: If the current word is nominal and the next word is PREP then the feature 'nomianlPREP' of the current word is set to '+1', otherwise, it is set to '-1'.

•Named Entity Information: The NE tag(s) of the previous word(s) can also be considered as the feature. This is the only dynamic feature in the experiment.

•First word: If the current token is the first word of a sentence, then the feature 'FirstWord' is set to '+1'; Otherwise, it is set to '-1'.

•Digit features: Several digit features have been considered depending upon the presence and/or the number of digit(s) in a token (e.g., ContainsDigit [token contains digits], FourDigit [token consists of four digits], TwoDigit [token consists of two digits]), combination of digits and punctuation symbols (e.g., ContainsDigitAndComma [token consists of digits and comma], ConatainsDigitAndPeriod [token consists of digits and periods]), combination of digits and symbols (e.g., ContainsDigitAndSlash [token consists of digit and slash], ContainsDigitAndHyphen [token consists of digits and hyphen], ContainsDigitAndPercentage [token consists of digits and percentages]). These binary valued features are helpful in recognizing miscellaneous NEs such as time expressions, monetary expressions, date expressions, percentages, numerical numbers etc.

•Gazetteer Lists: Various gazetteer lists have been developed from the partially NE tagged Bengali news corpus (Ekbal and Bandyopadhyay, 2007d). These lists have been used as the binary valued features of the SVM framework. If the current token is in a particular list, then the corresponding feature is set to '+1' for the current and/or surrounding word(s); otherwise, it is set to '-1'. The following is the list of gazetteers:

(i). Organization suffix word (94 entries): This list contains the words that are helpful in identifying organization names (e.g., *kong*, *limited* etc.). The feature 'OrganizationSuffix' is set to '+1' for the current and the previous words.

(ii). Person prefix word (245 entries): This is useful for detecting person names (e.g., *sriman*, *sree*, *srimati* etc.). The feature 'PersonPrefix' is set to '+1' for the current and the next two words.

(iii). Middle name (1,491 entries): These words generally appear inside the person names (e.g., *chandra*, *nath* etc.). The feature 'MiddleName' is set to '+1' for the current, previous and the next words.

(iv). Surname (5,288 entries): These words usually appear at the end of person names as their parts. The feature 'SurName' is set to '+1' for the current word.

(v). Common location word (547 entries): This list contains the words that are part of location names and appear at the end (e.g., *sarani*, *road*, *lane* etc.). The feature 'CommonLocation' is set to '+1' for the current word.

(vi). Action verb (221 entries): A set of action verbs like *balen*, *ballen*, *ballo*, *shunllo*, *haslo* etc. often determines the presence of person names. The feature 'ActionVerb' is set to '+1' for the previous word.

(vii). Frequent word (31,000 entries): A list of most frequently occurring words in the Bengali news corpus has been prepared using a part of the corpus. The feature 'RareWord' is set to '+1' for those words that are not in this list.

(viii). Function words (743 entries): A list of function words has been prepared manually. The feature 'NonFunctionWord' is set to '+1' for those words that are not in this list.

(ix). Designation words (947 entries): A list of common designation words has been prepared. This helps to identify the position of the NEs, particularly person names (e.g., *neta*, *sangsad*, *kheloar* etc.). The feature 'DesignationWord' is set to '+1' for the next word.

(x). Person name (72, 206 entries): This list contains the first name of person names. The feature 'PersonName' is set to '+1' for the current word.

(xi). Location name (7,870 entries): This list contains the location names and the feature 'LocationName' is set to '+1' for the current word.

(xii). Organization name (2,225 entries): This list contains the organization names and the feature 'OrganizationName' is set to '+1' for the current word.

(xiii). Month name (24 entries): This contains the name of all the twelve different months of both

English and Bengali calendars. The feature 'MonthName' is set to '+1' for the current word.
(xiv). Weekdays (14 entries): It contains the name of seven weekdays in Bengali and English both. The feature 'WeekDay' is set to '+1' for the current word.

4 Experimental Results

A partially NE tagged Bengali news corpus (Ekbal and Bandyopadhyay, 2007d) has been used to create the training set for the NER experiment. Out of 34 million wordforms, a set of 150K wordforms has been manually annotated with the 17 tags as shown in Table 1 with the help of *Sanchay Editor*[4], a text editor for Indian languages. Around 20K NE tagged corpus is selected as the development set and the rest 130K wordforms are used as the training set of the SVM based NER system.

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

$$P(t_1, t_2, t_3 ..., t_n \mid w_1, w_2, w_3 ..., w_n) = \prod_{i=1...n} P(t_i, w_i)$$

In this model, each word in the test data is assigned the NE tag that occurs most frequently for that word in the training data. The unknown word is assigned the NE tag with the help of various gazetteers and NE suffix lists.

Seventy four different experiments have been conducted taking the different combinations from the set 'F' to identify the best-suited set of features for NER in Bengali. From our empirical analysis, we found that the following combination gives the best result for the development set.

F={ $w_{i-3}w_{i-2}w_{i-1}w_iw_{i+1}w_{i+2}$, |prefix|<=3, |suffix|<=3, NE information of the window [-2, 0], POS information of the window [-1, +1], nominalPOS of the current word, nominalPREP, FirstWord, Digit features, Gazetteer lists}

The meanings of the notations, used in experimental results, are defined below:

pw, cw, nw: Previous, current and the next word; pwi, nwi: Previous and the next ith word from the current word; pt: NE tag of the previous word; pti: NE tag of the previous ith word; pre, suf: Prefix and suffix of the current word; ppre, psuf: Prefix and suffix of the previous word; npre, nsuf: Prefix and suffix of the next word; pp, cp, np: POS tag of the previous, current and the next word; ppi, npi: POS tag of the previous and the next ith word; cwnl: Current word is nominal.

Evaluation results of the development set are presented in Tables 2-4.

Feature (word, tag)	FS (%)
pw, cw, nw, FirstWord	71.23
pw2, pw, cw, nw, nw2, FirstWord	73.23
pw3, pw2, pw, cw, nw, nw2, FirstWord	**74.87**
pw3, pw2, pw, cw, nw, nw2, nw3, FirstWord	74.12
pw4, pw3, pw2, pw, cw, nw, nw2, FirstWord	74.01
pw3, pw2, pw, cw, nw, nw2, First Word, pt	75.30
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2	**76.23**
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, pt3	75.48
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, | |suf|<=4, pre|<=4	78.72
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3	**81.2**
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3 |psuf|<=3	80.4
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, |psuf|<=3, |nsuf|<=3, |ppre|<=3, |npre|<=3	78.14
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, |nsuf|<=3, |npre|<=3	79.90
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, |psuf|<=3, |ppre|<=3,	80.10
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit	**82.8**

Table 2. Results on the Development Set

It is observed from Table 2 that the word window [-3, +2] gives the best result (4th row) with the 'FirstWord' feature and further increase or decrease in the window size reduces the overall F-Score value. Results (7th-9th rows) show that the inclusion of NE information increases the F-Score value and the NE information of the previous two words gives the best results (F-Score=81.2%). It is indicative from the evaluation results (10th and 11th

[4]Sourceforge.net/project/nlp-sanchay

rows) that prefixes and suffixes of length up to three of the current word are very effective. It is also evident (12th-15th rows) that the surrounding word prefixes and/or suffixes do not increase the F-Score value. The F-Score value is improved by 1.6% with the inclusion of various digit features (15th and 16th rows).

Feature (word, tag)	FS (%)
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit, pp, cp, np	**87.3**
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit, pp2, pp, cp, np, np2	85.1
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit, pp, cp	86.4
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit, cp, np	85.8
pp2, pp, cp, np, np2, pt, pt2, |pre|<=3, |suf|<=3, FirstWord, Digit	41.9
pp, cp, np, pt, pt2, |pre|<=3, |suf|<=3, FirstWord, Digit	36.4
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit, cp	86.1

Table 3. Results on the Development Set

Experimental results (2nd-5th rows) of Table 3 suggest that the POS tags of the previous, current and the next words, i.e., POS information of the window [-1, +1] is more effective than the window [-2, +2], [-1, 0], [0, +1] or the current word alone. In the above experiment, the POS tagger was developed with 7 POS tags. Results (6th and 7th rows) also show that POS information with the word is helpful but only the POS information without the word decreases the F-Score value significantly. Results (4th and 5th rows) also show that the POS information of the window [-1, 0] is more effective than the POS information of the window [0, +1]. So, it can be argued that the POS information of the previous word is more helpful than the POS information of the next word.

In another experiment, the POS tagger was developed with 26 POS tags and the use of this tagger has shown the F-Score value of 85.6% with the feature (word, tag)=[pw3, pw2, pw, cw, nw, nw2, FirstWord, pt, pt2, |suf|<=3, |pre|<=3, Digit, pp, cp, np]. So, it can be decided that the smaller POS tagset is more effective than the larger POS tagset in NER. We have observed from two different experiments that the overall F-Score values can further be improved by 0.5% and 0.3%, respectively, with the 'nominalPOS' and 'nominalPREP' features. It has been also observed that the 'nominalPOS' feature of the current word is only helpful and not of the surrounding words. The F-Score value of the NER system increases to 88.1% with the feature: feature (word, tag)=[pw3, pw2, pw, cw, nw, nw2, FirstWord, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominalPREP].

Experimental results with the various gazetteer lists are presented in Table 4 for the development set. Results demonstrate that the performance of the NER system can be improved significantly with the inclusion of various gazetteer lists. The overall F-Score value increases to 90.7%, which is an improvement of 2.6%, with the use of gazetteer lists.

The best set of features is identified by training the system with 130K wordforms and tested with the help of development set of 20K wordforms. Now, the development set is included as part of the training set and resultant training set is thus consisting of 150K wordforms. The training set has 20,455 person names, 11,668 location names, 963 organization names and 11,554 miscellaneous names. We have performed 10-fold cross validation test on this resultant training set. The Recall, Precision and F-Score values of the 10 different experiments for the 10-fold cross validation test are presented in Table 5. The overall average Recall, Precision and F-Score values are 94.3%, 89.4% and 91.8%, respectively.

The other existing Bengali NER systems along with the *baseline* model have been also trained and tested with the same data set. Comparative evaluation results of the 10-fold cross validation tests are presented in Table 6 for the four different models. It presents the average F-Score values for the four major NE classes: 'Person name', 'Location name', 'Organization name' and 'Miscellaneous name'. Two different NER models, A and B, are defined in (Ekbal and Bandyopadhyay, 2007b). The model A denotes the NER system that does not use linguistic knowledge and B denotes the system that uses linguistic knowledge. Evaluation results of Table 6 show that the SVM based NER model has reasonably high F-Score value. The average F-Score value of this model is 91.8%, which is an improvement of 7.3% over the best-reported

HMM based Bengali NER system (Ekbal et al., 2007c). The reason behind the rise in F-Score value might be its better capability to capture the morphologically rich and overlapping features of Bengali language.

Feature (word, tag)	FS (%)
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominal-PREP, DesignationWord, Non-FunctionWord	89.2
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominal-PREP, DesignationWord, Non-FunctionWord	89.5
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominal-PREP, DesignationWord, Non-FunctionWord OrganizationSuffix, PersonPrefix	90.2
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominal-PREP, DesignationWord, Non-FunctionWord OrganizationSuffix, PersonPrefix, MiddleName, CommonLocation	90.5
pw3, pw2, pw, cw, nw, nw2, First Word, pt, pt2, |suf|<=3, |pre|<=3, Digit pp, cp, np, cwnl, nominal-PREP, DesignationWord, No-FunctionWord OrganizationSuffix, PersonPrefix, MiddleName, CommonLocation, Other gazetteers	**90.7**

Table 4. Results on the Development Set

The F-Score value of the system increases with the increment of training data. This fact is represented in Figure 1. Also, it is evident from Figure 1 that the value of 'Miscellaneous name' is nearly close to 100% followed by 'Person name', 'Location name' and 'Organization name' NE classes with the training data of 150K words.

Test set no.	Recall	Precision	FS (%)
1	92.5	87.5	89.93
2	92.3	87.6	89.89
3	94.3	88.7	91.41
4	95.4	87.8	91.40
5	92.8	87.4	90.02
6	92.4	88.3	90.30
7	94.8	91.9	93.33
8	93.8	90.6	92.17
9	96.9	91.8	94.28
10	97.8	92.4	95.02
Average	**94.3**	**89.4**	**91.8**

Table 5. Results of the 10-fold cross validation test

Model	F_P	F_L	F_O	F_M	F_T
Baseline	61.3	58.7	58.2	52.2	56.3
A	75.3	74.7	73.9	76.1	74.5
B	79.3	78.6	78.6	76.1	77.9
HMM	85.5	82.8	82.2	92.7	84.5
SVM	91.4	89.3	87.4	99.2	**91.8**

Table 6. Results of the 10-fold cross validation test (F_P: Avg. f-score of 'Person', F_L: Avg. f-score of 'Location', F_O: Avg. f-score of 'Organization', F_M: Avg. f-score of 'Miscellaneous' and F_T: Overall avg. f-score of all classes)

5 Conclusion

We have developed a NER system using the SVM framework with the help of a partially NE tagged Bengali news corpus, developed from the archive of a leading Bengali newspaper available in the web. It has been shown that the contextual window of size six, prefix and suffix of length up to three of the current word, POS information of the window of size three, first word, NE information of the previous two words, different digit features and the various gazetteer lists are the best-suited features for NER in Bengali. Experimental results with the 10-fold cross validation test have shown reasonably good Recall, Precision and F-Score values. The performance of this system has been compared with the existing three Bengali NER systems and it has been shown that the SVM-based system outperforms other systems. One possible reason behind the high Recall, Precision and F-Score values of the SVM based system might be its effectiveness to handle the diverse and overlapping features of the highly inflective Indian languages.

The proposed SVM based system is to be trained and tested with the other Indian languages, particularly Hindi, Telugu, Oriya and Urdu. Analyzing the performance of the system using other methods like MaxEnt and CRFs will be other interesting experiments.

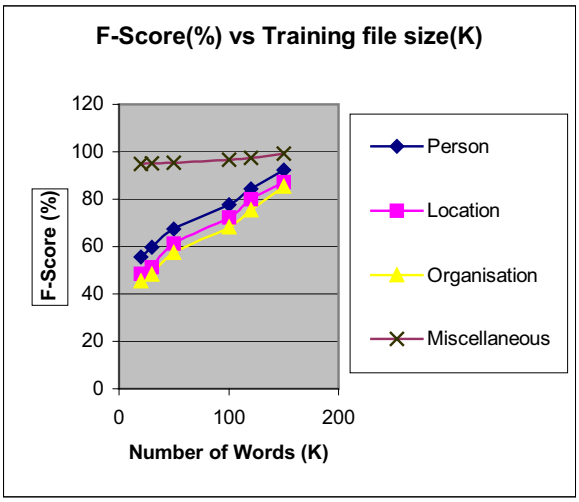

Fig. 1. F-Score VS Training file size

References

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

Asahara, Masayuki and Matsumoto, Yuji. 2003. Japanese Named Entity Extraction with Redundant Morphological Analysis. In *Proc. of HLT-NAACL.*

Babych, Bogdan, 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*, 1-8, Hungary.

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*, Maryland.

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 ICAPR*, India, 349-355.

Ekbal, Asif, and S. Bandyopadhyay. 2007b. Lexical Pattern Learning from Corpus Data for Named Entity Recognition. In *Proc. of ICON*, 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* (To appear December).

Joachims , T. 1999. Making Large Scale SVM Learning Practical. In *B. Scholkopf, C. Burges and A. Smola editions, Advances in Kernel Methods-Support Vector Learning, MIT Press*.

Kudo, Taku and Matsumoto, Yuji. 2001. Chunking with Support Vector Machines. In *Proceedings of NAACL,* 192-199.

Kudo, Taku and Matsumoto, Yuji. 2000. Use of Support Vector Learning for Chunk Identification. In *Proceedings of CoNLL-2000.*

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,* 282-289.

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.

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*, 1-10.

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

Takeuchi, Koichi and Collier, Nigel. 2002. Use of Support Vector Machines in Extended Named Entity Recognition. In *Proceedings of 6th CoNLL,* 119-125.

Vapnik, Valdimir N. 1995. The Nature of Statistical Learning Theory. *Springer*.

Yamada, Hiroyasu, Taku Kudo and Yuji Matsumoto. 2002. Japanese Named Entity Extraction using Support Vector Machine. In *Transactions of IPSJ,* Vol. 43, No. 1, 44-53.