2014 International Conference on Advances in ICT for Emerging Regions (ICTer): 215 - 220

Named Entity Recognition for Sinhala Language

J.K. Dahanayaka[#1], A.R. Weerasinghe[#2]

[#]*University of Colombo School of Computing,*
UCSC Building Complex, No. 35, Reid Avenue, Colombo 7, Sri Lanka.
[1]jinadikd@gmail.com
[2]arw@ucsc.cmb.ac.lk

***Abstract*— Named Entity Recognition (NER) is one of the major subtasks that have to be solved in most Natural Language Processing related tasks. However it is very much challenging to build a proper Named Entity Recognition system especially for Indic languages such as Sinhala because of the language features it inherits such as the absence of capitalization. Since there has not been much previous work based on NER for Sinhala, the concept and the needed resources have to be built from scratch. This paper tries to find out the effectiveness of using data-driven techniques to detect Named Entities in Sinhala text. Conditional Random Fields (CRF) and Maximum Entropy (ME) model were applied to this task. It is found that the former outperformed the latter in all experiments. A CRF model is able to detect Sinhala Named Entities with a very high precision (91.64%) and reasonable recall (69.34%) rates.**

***Keywords*— Conditional Random Fields, Maximum Entropy model, Named Entity, Named Entity Recognition, Natural Language Processing, Sinhala Language**

I. INTRODUCTION

Today with the vast growth of technology and information content, there is a need of retrieving the required information more efficiently out of the huge unstructured contexts in own native languages. To fulfill that need Natural Language Processing related research areas such as Information Extraction, Machine Translation, Information Retrieval and Automatic Summarization are essential. In all these areas Named Entity Recognition is one of the preliminary tasks that have to be performed.

The word 'Named' limits the task only to a word or a phrase which clearly identifies an entity over other entities having the same attributes in all possible worlds that it is referred in [1]. Named Entity (NE) does not have any strict agreed definition, but is explained according to the context it is used in. As an example, NEs are words or phrases which contain person names, organization names, countries, cities etc... Named Entity Recognition (NER) is the process of automatically identifying and classifying those entities into predefined Named Entity classes.

This paper discusses about recognizing NEs in Sinhala language, the native language of Sri Lanka which belongs to the Indo Aryan branch of the Indic language family, having approximately 19 million of speakers.

Although many successful, systems have been implemented for NER for English Language, the results of such research cannot be applied directly into Indic languages. The major obstacle is the absence of the capitalization feature. It has become more complex because of the following reasons [5], [2].

- Ambiguities about the words
- Free word-order
- Agglutinative nature
- Lack of resources
- Different ways of stating abbreviations
- Lack of standardization and variations in spellings

Even though NER research has been carried out for Indic languages such as Hindi and Bengali, Sinhala was not considered yet.

This work tries to present the effectiveness of using data-driven techniques (section IV) such as the Maximum Entropy model and Conditional Random Fields in detecting NEs in Sinhala text. The major contributions of this work are:

- A NE annotated corpus for Sinhala (described in section III.A) useful for future research
- A NE detection system (described in section IV) which would be applicable to Sinhala and other Indic languages.

A. Scope

In this research, only person names, location names and organization names are considered as Named Entities. The improved data driven solution that would be the outcome of this research is only capable in identifying whether the considered entity is a NE or not. Both training and testing datasets used should be in Sinhala Unicode.

II. RELATED BACKGROUND

NER is the task of segmenting and labelling sequence data. The approaches used can be categorized into three groups, namely; rule based methods, statistical or data-driven based methods and hybrid methods. Rule based approaches need accurate linguistics knowledge and analysis. Statistical approaches do not rely as much on this as rule based approaches, but need an annotated corpus. Hybrid systems use the strongest points of both methods.

The paper presented by Lisa F. Rau (1991) at the IEEE conference on artificial intelligence applications is considered as the first research paper in this field. It described a system to extract and recognize company names using heuristics and handcrafted rules. Till 1996, only a few publications can be found and most of them were based on English.

Recognizing the 'proper nouns' was the task done under the early works in NER. The three types of proper nouns categorized were person names, location names and organization names. Later this was extended to recognize more types.

In 1996, a huge significant framework was set up for NER at the 6th Message Understanding Conference (MUC- 6)[4]. From that conference onwards, NER is considered and practiced as one of the major sub tasks in Information Extraction.

Some of the major evaluation based events carried out for languages other than English are: MET-1998 (Multilingual

978-1-4799-7732-1/14/$31.00 ©2014 IEEE

Entity Tracking) project, IREX -1999 (Information Retrieval and Extraction Exercise), Shared task in CoNLL-2002, CoNLL-2003 and IJCNLP-08 workshop.

But in none of the above stated conferences and workshops was the Sinhala language considered. As languages inherit features from their language family it is believed that, there will be a higher probability of the applicability of the methods and techniques for Indic languages to Sinhala language too.

A. Evaluation Measurements

1) Precision (P): Fraction of correctly identified NEs which is relevant to total number of (both correct plus incorrect) NEs retrieved by the system that evaluated.

2) Recall (R): Fraction of the correctly identified NEs that is relevant to the total number of NEs in the reference data.

3) F-Measure: It is the weighted harmonic mean of precision and recall.

$$\text{F-Measure} = (\beta^2+1)\,PR / (\beta^2R+P)$$

β is the weighting between precision and recall typically $\beta=1$. When recall and precision are evenly weighted i.e. $\beta=1$, F-measure is also called F1-measure.

B. Named Entity Recognition for Indic Languages

NER for Indic languages received a major implies through the IJCNLP-08 workshop on NER for South and South East Asian languages [3]. It was focused on 5 Indic languages: Hindi, Bengali, Urdu, Oriya and Telugu [5]. It used 12 NE tags. A larger tagged set is used in aim of using the resultant NER model to be use for improving the performance of Machine Translation.

The data-driven techniques used for Indic languages are Hidden Markov model (HMM), Maximum Entropy (ME) model, Conditional Random Fields (CRF) and Support Vector Machines (SVM). Out of these the Maximum Entropy model and Conditional Random Fields can be seen to perform well for more languages.

The preferable features most of the authors have identified for Indic languages when using data-driven techniques are: word window of previous word and next word, suffix information (<=4 or <=3), previous NE tags, digit information, Part Of Speech (POS) based binary features, gazetteer lists, first word, word length and rare words [7],[13].

The value of F-measure obtained by [7] for Hindi when using the pure ME method is 75.89% and it was increased up to 81.12% when transliteration based gazetteers are added. The Hindi training dataset consists of 234K words collected from Hindi newspapers. This same method is applied for Bengali corpus with 64K words which gives out 69.59% F-measure value.

The best results came in [13] when using ME method recall, precision and F-measure values respectively for Bengali 88.01%, 82.63% and 85.22% and for Hindi 86.4%, 79.23% and 82.66% respectively. The reason of this higher score in Bengali is the use of language dependent features. Bengali training set had observed more number of NEs than Hindi training dataset even though the number of total words in Bengali training dataset is lower than Hindi training data set.

The paper [6] used the CRF model and the F-measure value obtained for Hindi, Bengali, Urdu, Oriya and Telugu respectively are 33.12%, 59.39%, 35.52%, 28.71%, 4.749%. Even though Hindi was having a large collection of training data the system failed to produce good results in the evaluation. And the poor performance of some of the languages is basically because use of the lack of training data.

Li and McCallum in [11] developed a successful NER system with the use of CRF with feature induction. They provide a large array of lexical tests and automatically discovered relevant features. With the use of feature induction, it was able to automatically construct features that increase the conditional likelihood most. It achieved a highest F-measure of 71.50% for Hindi.

C. Theoretical Background

1) Maximum Entropy Model: Maximum Entropy is a statistical model. The principle of Maximum Entropy states the correct distribution $p(a,b)$ which maximizes entropy or uncertainty [10]. It is having the notation of possible outputs (futures), histories, and features. This allows the computation of $p(f|h)$ for any f from space of possible futures F and for every h from the space of possible histories H [8]. In Maximum Entropy, a "history" is all of the conditioning data which enable to assign probabilities to the space of "futures" [8]. Probabilities are estimated in maximum entropy framework by using only a few assumptions and constraints.

In the task of Named Entity Recognition, we can form this concept as finding the probability of f combination with the token at index t in the test corpus. It can be expressed as, $p(f_t|h)$. To compute the $p(f|h)$, it needs a set of features, to make predictions about the futures. Features are binary functions of history and future [8].

The major advantage of this method, when comparing to Hidden Markov Model (HMM) is that it solves the problem of long distance dependency. However it suffers from 'label bias' problem.

2) Conditional Random Fields: Conditional Random Fields are based on the conditional probability framework. It is an undirected graphical model and matches up with conditionally trained probabilistic finite state automata [12]. CRF is capable of including arbitrary features easily, because they are conditionally trained. CRF has proven good results in practice on various tasks in segmenting and labelling sequence data.

CRF is having more advantages over other methods. It reduces the overhead of independence assumptions required by Hidden Markov Model and also avoids label bias problem [12] which showed as a problem in Maximum Entropy Markov Models. So CRF believed to be performing well rather than those Markov Models.

Lafferty *et al.* defined the probability of a particular label sequence $\mathbf{y}$ given observation sequence $\mathbf{x}$ to be the normalized product of potential functions [12].

III. SINHALA LANGUAGE MODEL

A. Corpus Building

Corpus is the main backbone for any data-driven technique. There is no NE tagged corpus prepared and

published for Sinhala which we can use directly. So we get the use of Part Of Speech (POS)-tagged corpus prepared by LTRL (Language Technology Research Laboratory) of UCSC (University of Colombo School of Computing). It consists of about 75,000 Sinhala words, collected from publisher content and archived web content. Each word and punctuation marks having the appropriate Part Of Speech tag label. It is ensure strictly not to use training data for testing. The training data set consist of about 68205 NE annotated Sinhala words while the test set has 5902 Sinhala words. That POS tagged corpus is retagged for maximal NEs and used as training dataset.

B. *Process Carried out*

The design process consists of the following steps:

- Preparing the NE annotated training dataset and un-annotated test dataset
- Extracting Sinhala language features from the training dataset
- Finding out customizable, freely downloadable open source ML toolkit for different data-driven methods
- Making the model files from the training data using feature vector
- Doing experiments with different feature sets
- Test using the test dataset
- Doing a comparative evaluation among the various data-driven methods used

To identify features from the corpus language specialized knowledge is essential and in Sinhala there are no features that can be identified directly and clearly. So it is a challenge to exact and select the suitable feature vector.

To implement the data-driven methods we take the use of two free and open source software toolkits. Those are Apache OpenNLP MaxEnt toolkit for Maximum Entropy model and CRF++ toolkit for Conditional Random Fields.

D. *Implementation and Feature Declaration*

1) Maximum Entropy Model: 'Apache OpenNLP MaxEnt' toolkit [14], version 1.6 is used to train and evaluate the Maximum Entropy based system for Sinhala. Start and End of a NE is only marked using special tags respectively as <START:NE> and <END>.

The features which we have tried out are described below,

- Context word feature – Words preceding and following a current word are considered as a feature for NER for Sinhala. Different experiments are done for various combinations of preceding and following words to improve its performance. This is also referred to as window size of the experiment.
- Word suffix – Suffixes are characters stripped from the end of the word.
- Language dependent feature – Most of the time, the word/words preceding මහතා, මයා, මිය, මෙනවිය, හිමි, පියතුමා are NEs.

2) *Conditional Random Fields*: We have tried out CRF++ [9], 0.58 version released in 2013. It is having multiple tokens (words) and each token having fixed number of multiple columns. Each token in a line is considered as a word in a sentence or a punctuation mark and columns are separated using spaces or a tab. When tagging the training set IOB tagging format was followed. It is used for tagging multiple NEs; denote the beginning word of a NE (B), inside word/words of a NE (I), and outside words (O).

Our testing and training datasets consists of 6 columns, the word itself, suffixes up to 4 and the answer tag.

Unlike Maximum Entropy, CRF does not need the careful selection of features because CRF can handle multiple arbitrary features without having the problem of over fitting. With the use of feature induction it can automatically construct the most useful feature combinations.

The features that have been tried out are,

- Context word feature
- Word suffix
- Bi-gram feature

IV. EVALUATION

The NER system has been trained with our prepared training data set. The evaluation is normally done comparing the system output with the output of human linguists. The test set is used to complete that task and it has not met any of the sentences in training dataset. Various experiments have been done changing the features in order to find the best suited feature set for detecting NEs in Sinhala text.

A. *Maximum Entropy Model*

The system was tested for default features in MaxEnt. Those are suffix length=1 and window size of one. Precision, recall and F-measure are 81.46%, 51.58% and 63.17% respectively. This is considered as the baseline for our experiments.

In the first series of experiments were done by changing the suffix length with window size of one (It considers previous word and next word from the focused word). Suffixes are characters stripped from the end of the word and changed this feature up to suffix length of 3. But surprisingly, it has not changed the resultant values which indicate us that suffix information is less important in ME for detecting NEs in Sinhala text.

The second set of experiments were done by including a language dependent feature as preceding entities of මහතා, මයා, මිය, මෙනවිය, හිමි, පියතුමා are NEs. When it used as a feature, show 81.71%, 51.34% and 63.06% for precision, recall and F-measure respectively. There is a slight increment in the value of the precision. When analyzing the datasets for the frequency of this kind of information, out of the total NEs appear in each dataset, the test set has 4.62% while training dataset contain 1.34%. This gives us a sense that these kinds of entities are not much frequently occurring

TABLE I
RESULTS OBTAIN IN ME FOR CONTEXT WORD FEATURE

Window size	Precision	Recall	F-measure
wp=1 wn=1	**81.71%**	**51.34%**	**63.06%**
wp=1 wn=2	78.21%	49.14%	60.36%
wp=1 wn=3	77.01%	51.59%	61.79%
wp=2 wn=1	75.81%	51.34%	61.22%
wp=2 wn=2	73.78%	48.17%	58.28%
wp=2 wn=3	73.06%	48.41%	58.24%
wp=3 wn=1	79.53%	49.39%	60.94%
wp=3 wn=2	75.10%	47.92%	58.51%
wp=3 wn=3	74.23%	47.19%	57.70%

wp = number of previous words from current focusing word
wn = number of next words from current focusing word

in both datasets. The probabilities are built using the frequency of occurrences in the training data set. If the training dataset is covering more percentage of these entities, it may improve the precision more.

Next set of experiments were done by changing the suffix length with language dependent features to see whether there will be any improvement. But surprisingly changes in suffixes have not affected much for the performance.

Finally the window size changed in both directions from the current focusing word. TABLE I clearly specified that best results came for when considering previous word and next word from the current focusing word. When the number of following words from the focusing word is increasing, precision recall and F-measure values reduce. It means that so much information on the following words do not provide any advantage in detecting NEs. When changing the number of preceding words from a current word, it reduces the recall value. Altogether from these results we can come to the conclusion that a lot of information about the surrounding entities from a current focusing entity does not provide any improvement in detecting NEs in Sinhala text using MaxEnt.

According to the results obtained for the various experiments conducted, we arrive at best suited feature vector for detecting NEs in Sinhala using ME is,

F = {w_{i-1} w_i w_{i+1} context word feature, language dependent features}

This feature vector was able to achieve precision, recall and F-measure respectively 81.71%, 51.34% and 63.06%.

B. Conditional Random Fields

Tagging a word in the test set when using this model, it concerns the probability of that word being in the training dataset. Unknown words take the use of feature vector and decide the suitable tag for the focusing word.

As a baseline system the current word and its corresponding tag is considered while using the bi-gram features. 83.50%, 40.63% and 54.66% are the values respectively obtained for precision recall and F-measure for this baseline system in CRF++ toolkit.

The first series of experiments were carried out by changing the context word feature. The best results achieved when the previous word, current word and next word combination was considered. Results are illustrated in TABLE II.

TABLE II
RESULTS OBTAIN IN CRF FOR CONTEXT WORD FEATURE

#	Feature vector		Precision	Recall	F-measure
1	B	cw	83.50%	40.63%	54.66%
2	**B**	**pw cw nw**	**91.91%**	**52.55%**	**66.87%**
3	B	pw_1 pw cw nw nw_1	89.79%	51.34%	65.33%
4	-	pw cw nw	52.21%	57.42%	54.69%

cw = current word, nw = next word, pw = previous word,
B = bi-gram feature, nw_i = next i^{th} word, pw_i= previous i^{th} word

When the window size is increasing equally in both directions from a focusing word, recall, precision and F-measure is going down. It means so much of information on surrounding words from the focusing word is not much beneficial in detecting NEs in Sinhala text. Even though higher number of NE predictions are done when bi-gram features are not used, but majority of the predicted NEs appear to be wrong. Thus it gives a sense that bi-gram features act a major role in detecting NEs in Sinhala text.

TABLE III
RESULTS OBTAIN IN CRF FOR CONTEXT WORD FEATURE WITH WORD COMBINATIONS

#	Feature vector		Precision	Recall	F-measure
1	B	cw	83.50%	40.63%	54.66%
5	**B**	**pw cw nw, com**	**93.22%**	**53.53%**	**68.01%**
6	B	pw_1 pw cw nw nw_1, com	89. 96%	50.12%	64.38%
7	-	pw cw nw, com	52.33%	57.42%	54.76%

cw = current word, nw = next word, pw = previous word,
B = bi-gram feature, nw_i = next i^{th} word, pw_i= previous i^{th} word

CRF is capable of giving arbitrary combination of various features. As the second experiment series we tried the feature vector given in TABLE III with combination of words as a feature. The word combinations denoted by 'com.'. 'com' value for the experiments in 5, 6, 7 respectively are {pw/cw, cw/nw}, {pw1/pw, pw/cw, cw/nw, nw/nw1} and {pw/cw, cw/nw}. According to the evaluation results shown in TABLE III when the combination of words is concerned, the values of precision, recall and F-measure goes higher. But still pw cw nw feature combination having the better performance. And the experiments here onwards concern the word combination feature too even it is not stated for the simplicity.

Third series of experiments are done by changing the window size in both directions unequally from the current focused word. The results in TABLE IV clearly showed that changes in the window size unequal in both directions (context word feature) having not much impact to the final result.

TABLE IV
RESULTS OBTAIN IN CRF FOR CONTEXT WORD FEATURE (CHANGING WINDOW SIZE UNEQUALLY IN BOTH DIRECTIONS) WITH WORD COMBINATIONS

#	Feature vector		P (%)	R(%)	F(%)
5	B	pw cw nw	93.22	53.53	68.01
8	B	pw cw nw nw_1	92.17	51.58	66.15
9	B	pw cw nw nw_1 nw_2	89.61	50.36	64.49
10	B	pw_1 pw cw nw	92.67	52.31	66.87
11	B	pw_1 pw cw nw nw_1	89.70	50.85	64.91
12	B	pw_1 pw cw nw nw_1 nw_2	88.94	48.91	63.11
13	B	pw_2 pw_1 pw cw nw	90.78	47.93	62.74
14	B	pw_2 pw_1 pw cw nw nw_1	88.69	47.69	62.03
15	B	pw_2 pw_1 pw cw nw nw_1nw_2	87.56	46.23	60.51

P = precision, R = recall, F = F-measure
cw = current word, nw = next word, pw = previous word,
B = bi-gram feature, nw_i = next i^{th} word, pw_i= previous i^{th} word

However, the best results were still obtained when the previous word, current word and next word are considered.

Too much information about the surrounding words is not presents good results, always it showed better results when the feature vector is as simple as possible. This can be cause because of over fitting of the features.

The next series of experiments was done by changing the suffix length, alone with the best results obtained in previous experiment series. Here, we have considered fixed length suffixes, fixed number of characters stripped out from the end of the current focusing word. In here, we have considered suffix length from 1 to 4 characters. Majority of the Sinhala words are equal or are less than 4 characters, so

TABLE V
RESULTS OBTAIN IN CRF FOR FIXED SUFFIX LENGTH FEATURE

#	Feature vector		Precision	Recall	F-measure
16	B	pw cw nw, suffix1 .	88.60%	58.64%	70.57%
17	B	pw cw nw, suffix2	91.21%	60.58%	72.81%
18	B	pw cw nw, suffix3	92.42%	59.37%	72.30%
19	B	pw cw nw, suffix4	92.89%	57.18%	70.78%

cw = current word, nw = next word, pw = previous word,
B = bi-gram feature,

considering more than 4 characters is not necessary and it becomes an overhead.

According to the results on TABLE V, higher recall and F-measure achieved on experiment no 17, considering {pw cw nw, suffix2} feature vector. It showed that when the suffix length increasing, precision increase while recall is decreasing. It means that the correct entities from the predicted entities are going higher with the increase of the suffix length.

Unlike ME, CRF is capable of handling combination of features. So the next series of experiments are done considering current word combined with the respective suffix. i.e., when current word combined with suffix1 denote as cw/suffix1.

TABLE VI
RESULTS OBTAIN IN CRF FOR SUFFIX LENGTH FEATURE WITH COMBINATION OF CURRENT WORD

#	Feature vector		P (%)	R (%)	F (%)
20	B	pw cwnw, suffix1, cw/suffix1	88.65	60.83	72.15
21	B	pw cwnw, suffix2, cw/suffix2	91.07	62.04	73.81
22	B	pw cwnw, suffix3, cw/suffix3	92.86	60.10	72.97
23	B	pw cwnw, suffix4, cw/suffix4	93.31	57.66	71.28

P = precision, R = recall, F = F-measure, B = bi-gram feature,
cw = current word, nw = next word, pw = previous word,

As stated in TABLE VI, the performance is going higher when the combination of current word and its relative suffix concerned. But still the higher performance showed when the suffix length is equal to 2. We consider more about the recall increment as it is the real value that showed the correctly predicted entities from the total number of entities appear in the test set. It improves the performance obtained in the previous set of experiments. The best results are obtained when the prefix size is exactly 2 and combine the suffix2 with the current focused word. Also we can see that these suffixes are not meaningful most of the time.

TABLE VII
RESULTS OBTAIN IN CRF FOR COMBINATION OF SUFFIX INFORMATION

#	Feature vector		P (%)	R (%)	F (%)
24	B	pw cw nw, |suffix|<=2	90.85	67.64	77.55
25	B	pw cw nw, |suffix|<=3	91.64	69.34	78.95
26	B	pw cw nw, |suffix|<=4	89.84	68.84	77.96

P = precision, R = recall, F = F-measure, B = bi-gram feature,
cw = current word, nw = next word, pw = previous word,

The next set of experiments is done to check whether combinations of this suffix information provide any improvement to the prevailing NER system. It is stated in TABLE VII. Suffix length equal to 2 showed better performance when considering individually suffixes. It is proven that when the |suffix length|<=3 the system performs well. But lots of information does not provide any better performance as it may cause over fittings.

By all these experiments finally we can come to a state regarding the best suiting feature vector for detecting NEs in Sinhala text using CRF method. That feature vector is

F= {$w_{i-1}w_i$ w_{i+1} context word feature, |suffix|<=3, Bi-gram features}.

This feature vector was able to detect 285 NEs out of 411 NE phrases, while obtaining precision, recall, F-measure values respectively 91.64%, 69.34% and 78.95%.

C. *Error Analysis*

The evaluation of NER systems is done against human performance. Thus, it is believed that human performance is correct at all times. Each and every word output came from each of the data-driven technique with the best performed feature vector, analyzed against human annotated output. The errors can be classified in to three categories such as,

- System predicted an entity where there is none.
- System noticed an entity but with wrong boundary.
- Completely missed by the system.

There are total 5902 words in the test set of which 411 were NEs. For each technique considering the output, the number of error entities that appear are identified and classified in to above error categories. Then the error rate is calculated for each error category.

TABLE VIII
ERROR RATES FOR EACH DATA-DRIVEN METHOD FOR ITS BEST PERFORMED FEATURE VECTOR

Error type	Error rate (%)	
	ME	CRF
Predicted but none	1.95	1.46
Identified with wrong boundary	8.03	4.85
Completely missed	49.64	25.79
Total error	59.61	32.12

According to TABLE VIII, a higher total error rate was recorded from ME method rather than CRF method and for each category ME error rate is higher. Both methods showed higher error rate in the completely missed entities error category, but CRF again performs better on that. Low recall in ME method appeared because of this higher error rate on completely missed entity category.

Both methods having the lowest error in the error category of system predicted an entity where there is none. Because of this low error value, both methods could achieve higher precision (81.71% and 91.64% for ME and CRF respectively).

Completely missed NEs error category having the highest error rate and it directly affects the recall. Thus to increase the recall, the entities in that category have to be closely considered. Some of the word entities can be seen in both models as completely missed. We have considered those overlapping words. There are about 85 entities that are overlapped in completely missed category. From the total NE count it is about 20.68%. Out of those 85 overlapped entities 42.35% are foreign person names and 34.12% are names of countries, local and foreign cities. The remaining are local person names and organization names. Most of these overlapped words can be identified with the use of gazetteer lists that are prepared lists of possible countries, cities, and organization names.

V. CONCLUSION

Named Entity Recognition (NER) is a major concern as one of the preliminary tasks that has to be done in many Natural Language Processing (NLP) related tasks such as Information Extraction, Machine Translation, Information Retrieval, and Automatic Summarization. It is essential to fulfill the need of accessing and retrieving the available information content efficiently and effectively on native languages other than English. Even though far matured NER solutions can be found for English, one cannot use them directly in Indic languages such as Sinhala owns to the absence of the capitalization feature. This paper tries to find out the effectiveness of using data-driven techniques in detecting NEs in Sinhala text.

As there is no published previous work targeting solving NER problem in Sinhala language, we make use of previous work that has been published for various Indic languages. It is applicable because languages inherit features from the language family it belongs to. Here we tried out two popular data-driven techniques that have been used frequently for Indic languages, namely Conditional Random Fields and Maximum Entropy model. Various experiments were carried out changing the feature set in order to find the best feature set for each model.

According to the results the most important and major conclusion that can be derived is that statistical modelling can be applied to NER for the Sinhala language. The most suitable feature set that can be beneficial in detecting NEs in Sinhala text is window size of one, suffix information and bi-gram features. It showed that Conditional Random Fields out performs Maximum Entropy model in this research.

Since the training set is more biased towards foreign news category, to take the full benefit from the NER system, the model should incorporate with sufficiently large, balanced corpus. It facilitates the model to make the learning process more efficient and accurate.

A. Future Work

Some of the future enhancements can be listed as follows,

- Prepare a sufficiently large and balanced corpus
- Develop more concise NE classes set which the final output will be beneficial in Machine Translation and other NLP related areas.
- Develop a NER system which is capable of classifying NEs into appropriate NE class.
- Do experiments concerning other data-driven techniques such as Support Vector Machines.
- Develop more precise algorithms using Hybrid approach with the use of gazetteer lists, morphological analyzer and language dependent rules.

ACKNOWLEDGMENT

Many thanks to all the wonderful people who have committed their lives for research in these areas and in particular to staff members of the LTRL in UCSC.

REFERENCES

[1] S. A. Kripke, *Naming and Necessity*, Harvard University Press, 1980.
[2] P. Sharma, U. Sharma and J. Kalita, "Named Entity Recognition: A Survey for the Indian Languages," *Language in India*, vol.11, May 2011.
[3] A.K. Singh, "Named Entity Recognition for South and South East Asian Languages:Taking Stock," *proc. 3rd Int. Conf. for Natural Language Processing workshop on NER for South and South East Asian Languages*, January, 2008, Hydrabad, India, pp. 5-16.
[4] R. Grishman and B. Sundheim, "Message Understanding Conference-6: A Brief History," *proc. 16th Int. Conf. on computational linguistics*, 1996, Copenhagen, Denmark, pp. 466-471.
[5] P. Kumar and R. Kiran, " A Hybrid Named Entity Recognition System for South Asian Languages," *proc. 3rd Int. Conf. for Natural Language Processing workshop on NER for South and South East Asian Languages*, January, 2008, Hydrabad, pp. 83-88.
[6] A. Ekbal, R. Haque, A.Das, V. Poka and S. Bandyopadhyay, "Language Independent Named Entity Recognition in Indian Languages," *proc. 3rd Int. Conf. for Natural Language Processing workshop on NER for South and South East Asian Languages*, January, 2008, Hydrabad, pp 33-40.
[7] S. K. Saha, P.S. Ghosh, S. Sarkar and P. Mitra, "Named Entity Recognition in Hindi using Maximum Entropy and Transliteration," *J. Polobits*, July-Dec. 2008, Issue 38, pp 33-42.
[8] A. E. Bothwick, "A Maximum Entropy Approach to Named Entity Recognition," Ph.D Dissertation, Dept. Compt. Sci., New York Univ, USA, 1999.
[9] (2014) "CRF++ homepage". [online]. Available: http://crfpp.googlecode.com/svn/trunk/doc/index.html
[10] A. Ratnaparkhi, "A Simple Information to Maximum Entropy Models for Natural Language Processing," Institute for research in Cognitive Science, Univ. of Pennsylvania, Tech. Rep. 97-08, 1997.
[11] W. Li and A. McCallum, "Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol.2, no.3, 2003, pp 290-294.
[12] J. D. Lafferty, A. McCallum and F.C.N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," *18th Int. Conf. on Machine Learning*, Williams College, Williamstown, MA, USA, June 28 – July 1, 2011, pp. 282-289.
[13] M. Hasanuzzaman, A. Ekbal and S. Bandyopadhyay, "Maximum Entropy Approach for Named Entity Recognition in Bengali and Hindi," *Int. J. of Recent trends in engineering*, vol.1, no.1, 2009, pp 589-594.
[14] (2014) "Apache OpenNLP MaxEnt". [online] Available: http://maxent.sourceforge.net