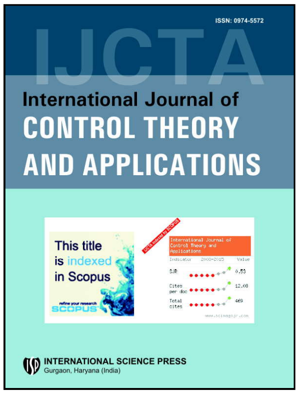

International Journal of Control Theory and Applications

ISSN : 0974-5572

© International Science Press

Volume 10 • Number 18 • 2017

Hybrid Multilingual Named Entity Recognition for Indian Languages

Sitanath Biswas[1] **and Sujata Dash**[1]

[1] *North Orissa University, Email: Sitanathbiswas2006@gmail.com,* sujata238dash@gmail.com

Abstract: Named Entity Recognition (NER) is an important task in natural language processing applicable to information extraction (IE), Machine Translation (MT), Information Retrieval (IR), Question Answering etc. NER is the task to identify and classify all proper nouns in a given text as person name, location name, organization name, number, time etc. Multilingual NER is a task where Named Entities can be recognized for various Languages by implementing one or more computation techniques. In this paper, we have used Conditional Random Field (CRF) as a base technique and Genetic Algorithm (GA) to effectively combine different feature representation. For better performance of this system, we have combined both the methods. We have taken three Indian languages Bengali, Hindi and Odiya for implementation. A very promising result is observed for all three languages while implementing GA with CRF.

Keywords: Genetic Algorithm, Conditional Random Field, Named Entity Recognition, Information Extraction, Machine Translation, Named Entity Recognition, Information Retrieval, Multilingual NER

1. INTRODUCTION

The increasing diversity of languages on the web introduced a new kind of complexity to Information Retrieval (IR) and Machine Translation (MT) systems. Named Entity Recognition (NER) is an important task in natural language processing pertaining to information extraction (IE), Machine Translation (MT), Information Retrieval (IR), Question Answering etc (Biswas, et. al 2011). NER is an active area of research for past twenty five years. A lot of progress has been made in named entities but NER still remains a big problem when it comes to Multilingual Named Entity Recognition task.

NER is a task of identifying and classifying all proper nouns in a given text as person name, location name, organization name, number time etc. As per the specifications given by Message Understanding Conference (MUC), the NER tasks normally works on seven types of named entities as listed below with their respective markup: (Biswas, et. al 2010)

PERSON (ENAMEX), ORGANISATION (ENAMEX), LOCATION (ENAMEX), DATE (TIMEX), TIME (TIMEX), MONEY (NUMEX), PERCENT (NUMEX)

A multilingual NER system is accountable for recognizing named entities in a wide variety of languages. In multilingual named entity recognition (NER), same method can be used for so many different types of languages and the extension of new languages is very easy and fast. There are several challenge to develop Multilingual NER system for Indian Languages is:

Indian languages belong to different language families, the major ones belongs to the Indo-European languages, Indo-Aryan and the Dravidian languages.

Morphologically rich - identifying root word is very difficult, requires use of morphological analysers.

No Capitalization feature like English—it is not found in Indian languages.

Ambiguity – Indian languages are ambiguities particularly common and proper nouns.

Spell variations –Same name can be spelled differently in different places on web.

In this paper we have implemented GA to search the correct feature selection. We have considered here the Orthography features, suffix and prefix information, morphology information, part-of-speech information as well as information about the surrounding words and their tags in Oriya language (Eqbal et.al 2009). We have used gazetteers for identification of designation, title, of the person names etc. We have also used person and location name gazetteers in our system for better identification of NEs. Linguistic rule also plays a crucial role in identifying NEs so we have used a number of linguistic rules in our system like the rule to recognize time, number etc. We have used CRF as base classifier because it is language independent and needs less computing overhead. The proposed approach is evaluated for three Indian languages Bengali, Hindi and Odiya. Hindi is the third most spoken language in the world and the national language of India. Bengali is the second popular language in India and the national language of Bangladesh. We have chosen Odiya language because significant work is not done for this language.

The rest of the paper is organized as follows. Section-2 introduces Conditional Random Field (CRF). Section-3 describes Genetic Algorithm (GA). Section-4 describes the proposed approach. Section-5 gives experimental results and section-6 gives conclusion.

2. CONDITIONAL RANDOM FIELD (CRF)

Conditional Random Fields (CRFs) are undirected graphical models, it is a special case of which corresponds to conditionally trained probabilistic finite state automata. As 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 and table extraction.(Eqbal et.al 2007) 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 = (s1, s2. . . sT) given an observation sequence o = (o1, o2, . . . , oT) is calculated as:

$$P_0\left(so\right)=\exp\left(\sum_{t=1}^{T}\sum_{k=1}^{K}kXf_k\left(S_t-1,S_{t,}o,t\right)\right),$$

where fk(st – 1 , st , o, t) is a feature function whose weight λk is learned through the training. The values of the feature functions can be between –∞... + ∞, but generally they are binary. To make all conditional probabilities sum up to 1, we have to calculate the normalization factor,

$$Z_0=\sum_{s}\exp\left(\sum_{t=1}^{T}\sum_{k=1}^{K}kXf_k\left(S_t-1,S_t,o,t\right)\right),$$

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

$$L_0 = \sum_{i=1}^{N} \log\left(P_0\left(s^{(i)} I o^{(i)}\right)\right) - \sum_{k=1}^{K} \frac{{}^{2}_{k}}{2\sigma^2},$$

where {o(i), s(i)} is the labelledtraining data. The second sum corresponds to a zero-mean,σ2 -variance Gaussian prior over parameters, that facilitates optimization by making the likelihood surface strictly convex. Here, we set parameters λ to maximize the penalized log-likelihood using limited-memory BFGS, a quasi-Newton method that is significantly more efficient, which results in only minor changes in accuracy due to changes in λ. When we apply 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 fk (st – 1, st, o, t) has a value of 0 for most cases and is only set to be 1, when st – 1, st are certain states and the observation has certain properties. We have used the C++ based CRF++ package, a simple, customizable, and open source implementation of CRF for segmenting or labelling sequential data.

http://crfpp.sourceforge.net.

3. GENETIC ALGORITHM (GA)

Genetic algorithms (GAs) are optimization techniques. It is also called randomized search. These are supervised by the principles of evolution and natural genetics. GAs having a very large amount of implicit parallelism. GAs perform search in very complex, large and multimodal landscapes. It provides near-optimal solutions for objective or the fitness function of an optimization problem (Gabrys B, Ruta D 2006). In GAs the search space parameters are encoded in the form of strings called chromosomes. A collection of such type of chromosomes is called a population. Initially, a random population is created, which represents different points in the search space. The objective or fitness function is associated with each string which represents the degree of goodness of the string. Basing upon the principle of survival of the fittest, a few of the strings are selected. Each selected string is assigned a number of copies that go into the mating pool. Operators which are biologically inspired like crossover and mutation are applied on these strings to produce a new generation of strings. The selection, crossover and mutation process continue for a fixed number of generations It continues till the termination condition is satisfied.

4. PROPOSED APPROACH

The proposed approach will work for three Indian Languages namely Hindi, Bengali and Odiya. The approach has following conventional stages of GA.

Fitness Computation:

Initially, the F-measure values of all the CRF based classifiers are calculated using 3-fold cross validation for the available training data. Suppose, there are M number of classifiers and their overall F-measure values be Fi, i = 1 . . . M.

If the last bit of the chromosome is 0, then the combined score of a particular class for a particular word w is:

$$f(ci) = \sum_{m=1:M} I_{ci}\left(op(w,m)\right)$$

if the last bit of the chromosome is 1, then the combined score of a particular class for a particular word w is:

$$f(ci) = \sum F_m,$$

∀ m = 1 : M & op(w, m) = ci

Selection:

In this paper, we have used roulette wheel selection. Here, the fitness function associated with each chromosome is used to associate a probability of selection with each individual chromosome. If fi is the fitness of individual i in the population, its probability of being selected is

$$Pi = \frac{f_i}{\sum_J^N I^f j}$$

where, N is the number of individuals in the population.

Crossover:

The normal single point crossover is considered here. The expressions for crossover probabilities are computed as follows. Let, fmax is the maximum fitness value of the current population, f is the average fitness value of the population and f is the larger of the fitness values of the solutions to be crossed. Then the probability of crossover, µc, is calculated as:

$$\mu c = k_1 x \frac{\left(f_{\max} - f'\right)}{f_{\max} - f}, \quad if\ f' > f$$

$$\mu c = k_3, \quad if\ f' \leq f.$$

Here, k1 and k3 are two variables and the values of k1 and k3 are kept equal to 1.0. when fmax = f, then f = fmax and µc will be equal to k3.

Mutation:

The expression for mutation probability, µm, is given below:

$$\mu_m = k_2 X \frac{\left(f_{\max} - f\right)}{f_{\max} - f} \quad if\ f > f,$$

$$\mu_m = k_4 \quad if\ f \leq f$$

Here, k2 and k4 are two variables. Here, values of k2 and k4 are kept equal to 0.5. This adaptive mutation helps GA to come out of local optimum. When GA converges to a local optimum, i.e. when fmax “ f decreases, both ìc and ìm will be increased. For a solution with the maximum fitness value, both ìc and ìm are zero. The best solution in a population is transferred undisrupted into the next generation.

5. EXPERIMENTAL RESULTS

We have used IJCNLP-08 Shared Task data on South and South East Asian Languages (NERSSEAL) and also manual annotated data for Odiya.

Table 1
Data set

Language	*No. of words in training*	*No. of NEs in training*	*No. of words in test*	*No. of NEs in test*
Hindi	314,760	38031	34810	4421
Bengali	432,113	28122	8965	1089
Odiya	87,238	4970	8460	1178

Table 2
Overall result for Hindi

Model	*Recall (in %)*	*Precision (in %)*	*F-measure (in %)*
GA based approach	71.27	83.95	77.09
Baseline	71.15	81.53	75.99

Table 3
Overall result for Bengali

Model	*Recall (in %)*	*Precision (in %)*	*F-measure (in %)*
GA based approach	74.72	87.15	80.46
Baseline	62.39	80.63	70.35

Table 4
Overall result for Odiya

Model	*Recall (in %)*	*Precision (in %)*	*F-measure (in %)*
GA based approach	60.91	94.15	73.97
Baseline	50.89	91.55	65.42

6. CONCLUSIONS

In this paper, we have proposed a GA based technique for CRF based NER. Features have been encoded in a chromosome. The average F-measure value of the CRF classifier trained using the feature set encoded in a particular chromosome has been used as the fitness value of that particular chromosome. One most promising characteristic of our system is that it makes use of the features that are language independent in nature, and can be easily obtained for many languages. Here, we evaluated our proposed technique for three resource-constrained Indian languages, namely Hindi, Bengali and Odiya. Evaluation results the overall recall, precision and F-measure values of 71.27%, 83.95% and 77.09%, respectively for Hindi, 74.27%, 87.15% and 80.46%, respectively for Bengali and 60.91%, 94.15% and 73.97%, respectively for Odiya.

REFERENCES

[1] Borthwick Andrew, A Maximum Entropy Approach to Named Entity Recognition. Ph.D. thesis, Computer Science Department, New York University, 1999.

[2] Li Wei and McCallum Andrew, Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction, In ACM Transactions on Computational Logi, 2004

[3] S. Biswas, et. al., A hybrids Oriya named entity recognition System: Harnessing the power of rule, International Journal of Artificial Intelligence and Expert Systems (IJAE), Vol (1) Issue 1, 1-6, 2011.

[4] S. Biswas, et. al., “A two stage language independent named entity recognition for Indian Languages”, 2010, International journal for computer Science and Information Technology, Vol. 1(4) 2010, 285-289, 2010.

[5] Kumaran. and Bhattacharyya Pushpak, Named Entity Recognition in Hindi using MEMM. In Technical Report, IIT Bombay, India, 2006.

[6] Srihari R., Niu C. and Li W., A Hybrid Approach for Named Entity and Sub-Type Tagging. In Proceedings of the sixth conference on applied natural language processing, 2000.

[7] Vapnik VN, Statistical learning theory. Wiley, New York, 1998.

[8] Dimililer N, Varoglu E, Altýnçay H., Vote-based classifier selection for biomedical NER using genetic algorithms. In: Proc of 3rd Iberian conference on pattern recognition and image analysis (IbPRAI 2007), vol 4478, pp 202–209, 2007.

[9] Gabrys B, Ruta D., Genetic algorithms in classifier fusion. Application Soft Computing 6(4):337–347, 2006.

[10] Alba, E., Luque, G., & Araujo, L., Natural language tagging with genetic algorithms. Information Processing Letters, 100(5), 173–182, 2006.

[11] Bennet, S. W., Aone, C., & Lovell, C., Learning to tag multilingual texts through observation. In Proceedings of empirical methods of natural language processing (pp. 109–116). Providence, Rhode Island, 1997.

[12] Ekbal, A., & Bandyopadhyay, S., 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) (pp. 51–58). India, 2008.

[13] Ekbal, A., & Bandyopadhyay, S., A conditional random field approach for named entity recognition in Bengali and Hindi. Linguistic Issues in Language Technology (LiLT), 2(1), 1–44, 2009.

[14] Ekbal, A., Naskar, S., & Bandyopadhyay, S., Named entity recognition and transliteration in Bengali. Named Entities: Recognition, Classification and Use, Special Issue of Lingvisticae Investigationes Journal, 30(1), 95–114, 2007.

[15] Kool, A., Daelemans, W., & Zavrel, J., Genetic algorithms for feature relevance assignment in memory-based language processing. In Proceedings of the 2nd workshop on learning language in logic and the 4th conference on computational natural language learning (pp. 103–106). Association for Computational Linguistics, 2000.