Experiments in Telugu NER: A Conditional Random Field Approach

Praneeth M Shishtla, Karthik Gali, Prasad Pingali and Vasudeva Varma
{praneethms,karthikg}@students.iiit.ac.in,{pvvpr,vv}@iiit.ac.in
Language Technologies Research Centre
International Institute of Information Technology
Hyderabad, India

Abstract

Named Entity Recognition(NER) is the task of identifying and classifying tokens in a text document into predefined set of classes. In this paper we show our experiments with various feature combinations for Telugu NER. We also observed that the prefix and suffix information helps a lot in finding the class of the token. We also show the effect of the training data on the performance of the system. The best performing model gave an $F_{\beta=1}$ measure of 44.91. The language independent features gave an $F_{\beta=1}$ measure of 44.89 which is close to $F_{\beta=1}$ measure obtained even by including the language dependent features.

1 Introduction

The objective of NER is to identify and classify all tokens in a text document into predefined classes such as person, organization, location, miscellaneous. The Named Entity information in a document is used in many of the language processing tasks. NER was created as a subtask in Message Understanding Conference (MUC) (Chinchor, 1997). This reflects the importance of NER in the area of Information Extraction (IE). NER has many applications in the areas of Natural Language Processing, Information Extraction, Information Retrieval and speech processing. NER is also used in question answering systems (Toral et al., 2005; Molla et al., 2006), and machine translation systems (Babych and Hartley, 2003). It is also a subtask in organizing and retrieving biomedical information (Tsai, 2006).
The process of NER consists of two steps

- identification of boundaries of proper nouns.
- classification of these identified proper nouns.

The Named Entities(NEs) should be correctly identified for their boundaries and later correctly classified into their class. Recognizing NEs in an English document can be done easily with a good amount of accuracy(using the capitalization feature). Indian Languages are very much different from the English like languages.

Some challenges in named entity recognition that are found across various languages are: Many named entities(NEs) occur rarely in the corpus i.e they belong to the open class of nouns. Ambiguity of NEs. Ex *Washington* can be a person's name or a place name. There are many ways of mentioning the same Named Entity(NE). In case of person names, *Ex: Abdul Kalam, A.P.J.Kalam, Kalam* refer to the same person. And, in case of place names *Warangal, WGL* both refer to the same location. Named Entities mostly have initial capital letters. This discriminating feature of NEs can be used to solve the problem to some extent in English.

Indian Languages have some additional challenges: We discuss the challenges that are specific to Telugu. Absence of capitalization. Ex: The condensed form of the person name *S.R.Shastry* is written as *S.R.S* in English and is represented as *srs* in Telugu. Agglutinative property of the Indian Languages makes the identification more difficult. Agglutinative languages such as Turkish or Finnish, Telugu etc. differ from languages like English in

Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 105–110,
Hyderabad, India, January 2008.

the way lexical forms are generated. Words are formed by productive affixations of derivational and inflectional suffixes to roots or stems. *For example: warangal, warangal ki, warangalki, warangallo, warangal ni* etc .. all refer to the place Warangal. where *lo, ki, ni* are all postpostion markers in Telugu. All the postpositions get added to the stem hyderabad. There are many ways of representing acronyms. The letters in acronyms could be the English alphabet or the native alphabet. Ex: *B.J.P* and *BaJaPa* both are acronyms of *Bharatiya Janata Party*. Telugu has a relatively free word order when compared with English. The morpohology of Telugu is very complex. The Named Entity Recognition algorithm must be able handle most of these above variations which otherwise are not found in languages like English. There are not rich and robust tools for the Indian Languages. For Telugu, though a Part Of Speech(POS) Tagger for Telugu, is available, the accuracy is less when compared to English and Hindi.

2 Problem Statement

NER as sequence labelling task
Named entity recognition (NER) can be modelled as a sequence labelling task (Lafferty et al., 2001). Given an input sequence of words $W_1^n = w_1 w_2 w_3 ...w_n$, the NER task is to construct a label sequence $L_1^n = l_1 l_2 l_3 ...l_n$, where label l_i either belongs to the set of predefined classes for named entities or is none(representing words which are not proper nouns). The general label sequence l_1^n has the highest probability of occuring for the word sequence W_1^n among all possible label sequences, that is

$$\hat{L}_1^n = \text{argmax} \{\Pr(L_1^n \mid W_1^n)\}$$

3 Conditional Random Fields

Conditional Random Fields (CRFs) (Wallach, 2004) are undirected graphical models used to calculate the conditional probability of values on designated output nodes given values assigned to other designated input nodes. In the special case in which the output nodes of the graphical model are linked by edges in a linear chain, CRFs make a first-order Markov independence assumption, and thus can be understood as conditionally-trained finite state machines(FSMs).

Let o = $\langle O_1, O_2, ...O_T \rangle$ be some observed input data sequence, such as a sequence of words in text in a document,(the values on n input nodes of the graphical model). Let **S** be a set of FSM states, each of which is associated with a label, $l \in \mathscr{L}$.
Let **s** = $\langle s_1, s_2, ... s_T, \rangle$ be some sequence of states,(the values on T output nodes). By the Hammersley-Clifford theorem CRFs define the conditional probability of a state sequence given an input sequence to be

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

where Z_o is a normalization factor over all state sequences, is an arbitrary feature function over its arguments, and λ_k is a learned weight for each feature function. A feature function may, for example, be defined to have value 0 or 1. Higher λ weights make their corresponding FSM transitions more likely.

CRFs define the conditional probability of a label sequence based on total probability over the state sequences, $P(l|o) = \sum_{s:l(s)=l} P(s|o)$ where l(s) is the sequence of labels corresponding to the labels of the states in sequence s. Note that the normalization factor, Z_o, (also known in statistical physics as the partition function) is the sum of the scores of all possible state sequences,

$$Z_o = \sum_{s \in S^T} * exp(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(s_{t-1}, s_t, o, t))$$

and that the number of state sequences is exponential in the input sequence length,T. In arbitrarily-structure CRFs, calculating the partition function in closed form is intractable, and approximation methods such as Gibbs sampling, or loopy belief propagation must be used.

4 Features

There are many types of features used in general NER systems. Many systems use binary features i.e. the word-internal features, which indicate the presence or absence of particular property in the word. (Mikheev, 1997; Wacholder et al., 1997; Bikel et al., 1997). Following are examples of binary features commonly used. All-Caps (IBM), Internal capitalization (eBay), initial capital (Abdul Kalam), uncapitalized word (can), 2-digit number

(83, 28), 4-digit number (1273, 1984), all digits (8, 31, 1228) etc. The features that correspond to the capitalization are not applicable to Telugu. We have not used any binary features in our experiments.

Gazetteers are used to check if a part of the named entity is present in the gazetteers. We don't have proper gazetteers for Telugu.

Lexical features like a sliding window [w_{-2},w_{-1},w_o,w_1,w_2] are used to create a lexical history view. Prefix and suffix tries were also used previously(Cucerzan and Yarowsky,1999).

Linguistics features like Part Of Speech, Chunk, etc are also used.

4.1 Our Features

We donot have a highly accurate Part Of Speech(POS) tagger. In order to obtain some POS and chunk information, we ran a POS Tagger and chunker for telugu (PVS and G, 2007) on the data. And from that, we used the following features in our experiments.

Language Independent Features
current token: w_0
previous 3 tokens: w_{-3},w_{-2},w_{-1}
next 3 tokens: w_1,w_2,w_3
compound feature:w_0 w_1
compound feature:w_{-1} w_0
prefixes (len=1,2,3,4) of w_0: pre_0
suffixes (len=1,2,3,4) of w_0: suf_0

Language Dependent Features
POS of current word: POS_0
Chunk of current word: $Chunk_0$

Each feature is capable of providing some information about the NE.

The word window helps in using the context information while guessing the tag of the token. The prefix and suffix feature to some extent help in capturing the variations that may occur due to agglutination.

The POS tag feature gives a hint whether the word is a proper noun. When this is a proper noun it has a chance of being a NE. The chunk feature helps in finding the boundary of the NE.

In Indian Languages suffixes and other inflections get attached to the words increasing the length of the word and reducing the number of occurences of that word in the entire corpus. The character n-grams can capture these variations.

5 Experimental Setup

5.1 Corpus

We conducted the experiments on the developement data released as a part of NER for South and South-East Asian Languages (NERSSEAL) Competetion. The corpus in total consisted of 64026 tokens out of which 10894 were Named Entities(NEs). We divided the corpus into training and testing sets. The training set consisted of 46068 tokens out of which 8485 were NEs. The testing set consisted of 17951 tokens out of which 2407 were NEs. The tagset as mentioned in the release, was based on AUKBC's ENAMEX,TIMEX and NAMEX, has the following tags: NEP (Person), NED (Designation), NEO (Organization), NEA (Abbreviation), NEB (Brand), NETP (Title-Person), NETO (Title-Object), NEL (Location), NETI (Time), NEN (Number), NEM (Measure) & NETE (Terms).

5.2 Tagging Scheme

The corpus is tagged using the IOB tagging scheme (Ramshaw and Marcus, 1995). In this scheme each line contains a word at the beginning followed by its tag. The tag encodes the type of named entity and whether the word is in the beginning or inside the NE. Empty lines represent sentence(document) boundaries. An example is given in table 1.

Words tagged with O are outside of named entities and the I-XXX tag is used for words inside a named entity of type XXX. Whenever two entities of type XXX are immediately next to each other, the first word of the second entity will be tagged B-XXX in order to show that it starts another entity. This tagging scheme is the IOB scheme originally put forward by Ramshaw and Marcus (1995).

5.3 Experiments

To evaluate the performance of our Named Entity Recognizer, we used three standard metrics namely precision, recall and f-measure. Precision measures the number of correct Named Entities(NEs) in the

Token	Named Entity Tag
Swami	B-NEP
Vivekananda	I-NEP
was	O
born	O
on	O
January	B-NETI
,	I-NETI
12	I-NETI
in	O
Calcutta	B-NEL
.	O

Table 1: IOB tagging scheme.

machine tagged file over the total number of NEs in the machine tagged file and the recall measures the number of correct NEs in the machine tagged file over the total number of NEs in the golden standard file while F-measure is the weighted harmonic mean of precision and recall:

$$F = \frac{\left(\beta^2+1\right)\ RP}{\beta^2 R + P}$$

with

$$\beta = 1$$

where P is Precision, R is Recall and F is F-measure.

W_{-n+n}: A word window :w_{-n}, w_{-n+1}, .., w_{-1}, w_0, w_1, .., w_{n-1}, w_n.
POS_n: POS n^{th} token.
Ch_n: Chunk of n^{th} token.
pre_n: Prefix information of n^{th} token. (prefix length=1,2,3,4)
suf_n: Suffix information of n^{th} token. (suffix length=1,2,3,4)

The more the features, the better is the performance. The inclusion of the word window, prefix and suffix features have increased the $F_{\beta=1}$ measure significantly. Whenever the suffix feature is included, the performance of the system increased. This shows that the system is able to caputure those agglutinative language variations. We also have experimented changing the training data size. While varying the training data size, we have tested the performance on the same amount of testing data of 17951 tokens.

6 Conclusion & Future Work

The inclusion of prefix and suffix feature helps in improving the $F_{\beta=1}$ measure (also recall) of the system. As the size of the training data is increased, the $F_{\beta=1}$ measure is increased. Even without the language specific information the system is able to perform well. The suffix feature helped improve the recall. This is due to the fact that the POS tagger also uses the same features in predicting the POS tags. Prefix, suffix and word are three non-linguistic features that resulted in good performance. We plan to experiment with the character n-gram approach (Klein et al., 2003) and include gazetteer information.

References

Bogdan Babych and Anthony Hartley. 2003. Improving machine translation quality with automatic named entity recognition. In *Proceedings of Seventh International EAMT Workshop on MT and other language technology tools*, Budapest, Hungary.

Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Nancy Chinchor. 1997. Muc-7 named entity task definition. Technical Report Version 3.5, Science Applications International Corporation, Fairfax, Virginia.

Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. 2003. Named entity recognition with character-level models. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 180–183, Morristown, NJ, USA. Association for Computational Linguistics.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA.

Andrei Mikheev. 1997. Automatic rule induction for unknown-word guessing. *Comput. Linguist.*, 23(3):405–423.

Features	Precision	Recall	$F_{\beta=1}$
Ch_0	51.41%	9.19%	15.59
POS_0	46.32%	9.52%	15.80
$POS_0.Ch_0$	46.63%	9.69%	16.05
$W_{-3+3}.Ch_0$	59.08%	19.50%	29.32
$W_{-3+3}.POS_0$	58.43%	19.61%	29.36
$Ch_0.pre_n$	53.97%	24.76%	33.95
$POS_0.pre_n$	53.94%	24.93%	34.10
$POS_0.Ch_0.pre_n$	53.94%	25.32%	34.46
$POS_0.suf_n$	47.51%	29.36%	36.29
$POS_0.Ch_0.suf_n$	48.02%	29.24%	36.35
$Ch_0.suf_n$	48.55%	29.13%	36.41
$W_{-3+3}.POS_0.pre_n$	62.98%	27.45%	38.24
$W_{-3+3}.POS_0.Ch_0.pre_n$	62.95%	27.51%	38.28
$W_{-3+3}.Ch_0.pre_n$	62.88%	27.62%	38.38
$W_{-3+3}.POS_0.suf_n$	60.09%	30.53%	40.49
$W_{-3+3}.POS_0.Ch_0.suf_n$	59.93%	30.59%	40.50
$W_{-3+3}.Ch_0.suf_n$	61.18%	30.81%	40.98
$POS_0.Ch_0.pre_n.suf_n$	57.83%	34.57%	43.27
$POS_0.pre_n.suf_n$	57.41%	34.73%	43.28
$Ch_0.pre_n.suf_n$	57.80%	34.68%	43.35
$W_{-3+3}.Ch_0.pre_n.suf_n$	64.12%	34.34%	44.73
$W_{-3+3}.POS_0.pre_n.suf_n$	64.56%	34.29%	44.79
$W_{-3+3}.POS_0.Ch_0.pre_n.suf_n$	64.07%	34.57%	44.91

Table 2: Average Precision,Recall and $F_{\beta=1}$ measure for different language dependent feature combinations.

Features	Precision	Recall	$F_{\beta=1}$
w	57.05%	20.62%	30.29
pre	53.65%	23.87%	33.04
suf	47.75%	29.19%	36.23
w.pre	63.08%	27.56%	38.36
w.suf	60.93%	30.76%	40.88
pre.suf	57.94%	34.96%	43.61
w.pre.suf	64.80%	34.34%	44.89

Table 3: Average Precision,Recall and $F_{\beta=1}$ measure for different language independent feature combinations.

Diego Molla, Menno van Zaanen, and Daniel Smith. 2006. Named entity recognition for question answering. In *Proceedings of Australasian Language Technology Workshop 2006*, Sydney, Australia.

Avinesh PVS and Karthik G. 2007. Part-of-speech tagging and chunking using conditional random fields and transformation based learning. In *In Proceedings of SPSAL-2007 Workshop*.

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In David Yarovsky and Kenneth Church, editors, *Proceedings of the Third Workshop on Very Large Corpora*, pages 82–94, Somerset, New Jersey. Association for Computational Linguistics.

Antonio Toral, Elisa Noguera, Fernando Llopis, and Rafael Muñoz. 2005. Improving question answering using named entity recognition. In *Proceedings of the 10th NLDB congress*, Lecture notes in Computer Science, Alicante, Spain. Springer-Verlag.

Number of Words	Precision	Recall	$F_{\beta=1}$
2500	51.37%	9.47%	15.99
5000	64.74%	11.93%	20.15
7500	61.32%	13.50%	22.13
10000	66.88%	23.31%	34.57
12500	63.42%	27.39%	38.26
15000	63.55%	31.26%	41.91
17500	60.58%	30.64%	40.70
20000	58.32%	30.03%	39.64
22500	57.72%	29.75%	39.26
25000	59.33%	29.92%	39.78
27500	60.91%	30.03%	40.23
30000	62.77%	30.42%	40.98
32500	62.66%	30.64%	41.16
35000	62.08%	30.81%	41.18
37500	61.02%	30.87%	41.00
40000	61.60%	31.09%	41.33
42500	62.12%	32.44%	42.62
45000	62.70%	32.77%	43.05
47500	63.20%	32.72%	43.12
50000	64.29%	34.29%	44.72

Table 4: The effect of training data size on the performance of the NER.

Richard Tzong-Han Tsai. 2006. A hybrid approach to biomedical named entity recognition and semantic role labeling. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 243–246, Morristown, NJ, USA. Association for Computational Linguistics.

Nina Wacholder, Yael Ravin, and Misook Choi. 1997. Disambiguation of proper names in text. In *Proceedings of the fifth conference on Applied natural language processing*, pages 202–208, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Hanna M. Wallach. 2004. Conditional random fields: An introduction. Technical Report MS-CIS-04-21, University of Pennsylvania, Department of Computer and Information Science, University of Pennsylvania.