Conditional Random Fields for Code Mixed Entity Recognition

[NLP_CEN_AMRITA@CMEE-IL-FIRE-2016]

Barathi Ganesh HB
Artificial Intelligence Practice
Tata Consultancy Services
Kochi - 682 042
India
barathiganesh.hb@tcs.com

Anand Kumar M and Soman KP
Center for Computational Engineering and Networking (CEN)
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham
Amrita University, India
m_anandkumar@cb.amrita.edu,
kp_soman@.amrita.edu

ABSTRACT

Entity Recognition is an essential part of Information Extraction, where explicitly available information and relations are extracted from the entities within the text. Plethora of information is available in social media in the form of text and due to its nature of free style representation, it introduces much complexity while mining information out of it. This complexity is enhanced more by representing the text in more than one language and the usage of transliterated words. In this work we utilized sequential modeling algorithm with hybrid features to perform the Entity Recognition on the corpus given by CMEE-IL (Code Mixed Entity Extraction - Indian Language) organizers. The experimented approach performed great on both the Tamil-English and Hindi-English tweet corpus by attaining nearly 95% against the training corpus and 45.17%, 31.44% against the testing corpus.

Keywords

Entity Recognition; Sequential Modeling; Conditional Random Fields

1. INTRODUCTION

The information shared by people in this digital era has continuous growth in nature (facebook[1], twitter[2]). Mining information from these social media text has becomes essential for both the government and industrial sectors. Moreover these texts serves as a information source for different text applications [13].

Entity Recognition is one of the major key component in information extraction applications, which could be used to extract the implicitly and explicitly available information and relation between the information [8, 11]. Entity Recognition is a task of assigning words or phrases in a text into its predefined set of real world world entities like person, location, organization, ..., etc [10].

Due to the constraints introduced by the social media platforms (number of words and formats) and due to the absence of proper constraints in usage of shared text (grammar and in-proper words), mining information from social media text has become complex to achieve. When shared texts incorporates multiple languages and transliterated words, it introduces much complexity to make the fully automated analytics system [7]. So far text analytics applications focused on English text alone. In recent works, it can be observed that, researches started contributing towards the code mixed text analytics applications [12, 5, 3, 1].

By observing the above, we have experimented the sequential modeling algorithm - Conditional Random Fields (CRF) along with the hybrid features for performing entity extraction on code mixed social media texts (i.e. tweets). A set of corpus based lexicon features extracted out of the words in the tweets to make Random Forest Tree based binary classifier (Entity, Non- Entity). This classifier predicts the given word is entity or not. Along with this binary result, other common lexicon features are utilized to build the CRF based entity recognizer.

Remaining of the paper details about the CRF for entity recognition in section 2, Random Forest Tree as a binary classifier in section 3, feature engineering carried over in section 4 and section 5 details about the experimentation and observations about the results achieved.

2. SEQUENTIAL MODELING WITH CONDITIONAL RANDOM FIELDS

Over the last few years, CRF has became the pioneer algorithm in sequential modeling applications (Part Of Speech tagging, Named Entity Recognition) [9, 2]. CRF is from a discriminative and undirected-probabilistic graphical model, which is generally used in structured prediction application. Unlike other ordinary classification methods CRF has a capability of classifying sequence of sample (i.e. context loading with respect to the neighbouring words).

The advantages of CRF over other sequential modeling algorithms are it avoids the label biasing problem; conditional probability distribution made over the target label sequences (i.e. sequence of tags) given a input sequences (i.e. sequence of words); it has a capability to easily include a wide variety of arbitrary and non-independent features with respect to the input words [6]. Let $x_{1:N}$ be the word sequence and $y_{1:N}$ is the output label sequence, then the CRF can be mathematically represented as,

[1] www.facebook.com
[2] www.twitter.com

Feature Type	Binary	Nominal
Type of the Word All upper, All Digit, Alphanumeric word, All symbols, All letter First letter capital,	✓	
Shape of the Word ex:- (Vijay - Uuuuu 11-12-1991 - nnsnnsnnnn)	✓	✓
Part of Speech Tag		✓
Prefix of length 1 to 4 eg:- (Parking- g, ng, ing, king)		✓
Suffix of length 1 to 4 eg:- (Parking- P, Pa, Par, Park)		✓
Length of Word		✓
Entity or not Decision from Random Forest Tree Classifier	✓	

Table 1: CRF Features

$$\frac{1}{Z}\exp\left(\sum_{j}\lambda_j t_j\left(y_{i-1},y_i,x,i\right)+\sum_{k}\mu_k s_k\left(y_i,x,i\right)\right) \qquad (1)$$

$$Z=\sum_{y_{1:N}}\exp\left(\sum_{j}\lambda_j t_j\left(y_{i-1},y_i,x,i\right)+\sum_{k}\mu_k s_k\left(y_i,x,i\right)\right) \qquad (2)$$

In above equation x represents the input word sequence ([Vijay, acted, in, a, film, Sura]), y represents the output label sequence ([Actor, other, other, other, other, Entertainment]), $t_j\left(y_{i-1},y_i,x,i\right)$ is a transition function constrained by the feature function as given in equation 3 (i.e. probability of label changing from one label to another learned from training corpus and change of label at position $i-1$ to i in test sequence), $s_k\left(y_i,x,i\right)$ is similar to the emission probability at Hidden Morkov Model but constrained by feature function similar to $t_j\left(y_{i-1},y_i,x,i\right)$, Z is the normalization factor and λ_j, μ_k are the optimization parameters learned from training corpus.

The transition function $t_j\left(y_{i-1},y_i,x,i\right)$ and emission function $s_k\left(y_i,x,i\right)$ takes on the values only if $b(x,i)$ is greater than 0. $b(x,i)$ will be greater than 0, if the current state (in the case of the emission functions), previous and current states (in the case of the transition functions) take on particular values with respect to the training corpus. An example, $b(x,i)$ activation function is given below:

$$t_j\left(y_{i-1},y_i,x,i\right)=\begin{cases} b\left(x,i\right) & if y_{i-1}=other\ and \\ & y_i=Entertainment \\ 0 & otherwise \end{cases} \qquad (3)$$

In the above equation, $b(x,i)$ will be greater than 0 only if the following two labels (other, Entertainment) consecutively occurs in the training set. From the above inputs, it is clear that transition and emission functions are constrained with respect to the feature function $b(x,i)$. Incorporating relevant features from the training set will leads to a high performance sequential modeling system. Few of the nominal and binary features utilized in this proposed approach is given in the Table 1.

3. ENTITY SELECTION WITH RANDOM FOREST TREE

The feature mentioned in Table 1 i.e. Entity or not, is a binary function derived through Random Forest Tree classifier. More than the other features mentioned in the Table 1, this binary feature provide more constraint to the feature function in CRF to find distribution over the output label. Random Forest Tree is a classification algorithm, which is formed by selecting the most occurring resultant class among the set of weak decision trees [4]. In this approach the lexicon based features from the entity words are extracted and by considering these features as the attributes for the Random Forest Tree classifier, the classes (entity, not a entity) for the given word is predicted.

Given a training set $W=w1,w2,w3,...,wn$ (words) with the output labels $Y=y1,y2,y3,...,yn$ (entity , not a entity) and feature set $F=f1,f2,f3,...,fn$, bagging repeatedly (B times - Number of trees) done by selecting random samples and attributes from the training set and builds the decision tree for each set. Then the predictions for test words $\hat{W}$ can be found by averaging the predictions from all the individual decision trees built through the train set. It can be interpreted as following:

$$f_b=f(W_b,Y_b,F_b) \qquad (4)$$

$$Y=\frac{1}{B}\sum_{b=1}^{B}f_b(\hat{W}\hat{F}) \qquad (5)$$

Corpus based lexicon features are extracted in-order to train the above classifier. Initially a feature set is built from the entity words available in Tamil-English and Hindi-English corpus. Then by taking these features as a vocabulary, the Term - Document Matrix (TDM) is built against the words. Then this matrix along with the binary labels (entity, not a entity) are fed to the Random Forest Tree to make the decision. The feature set of TDM includes prefix and suffix of length 1 to 3 of the words, length of words and position of the word in that tweet.

Discription	Tamil-English	Hindi-English
# Tweets	3184.0	2701.0
# Unique Tweets	2821.0	2669.0
# Tags	1624.0	2413.0
# Unique Tags	21.0	21.0
# Entity words	1624.0	2413.0
# Unique Entity words	1016.0	1200.0
# words	32142.0	43766.0
Avg # words / tweet	10.1	16.2
Entity-Word ratio	5.1%	5.5%

Table 2: Data-set Statistics

4. EXPERIMENT AND OBSERVATIONS

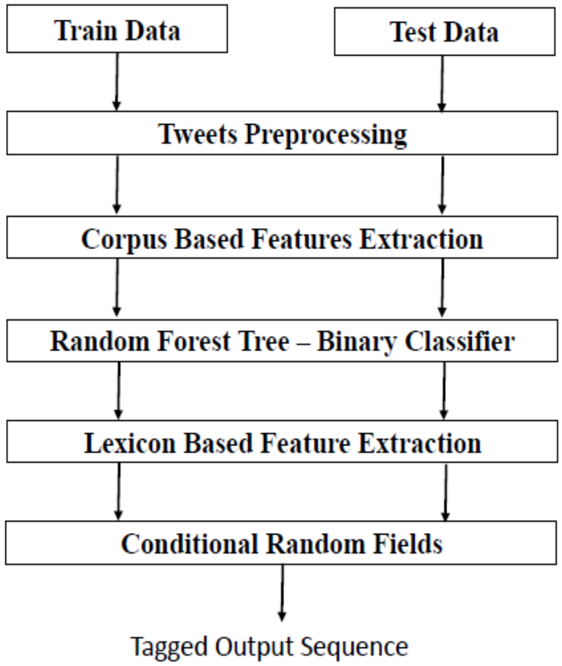
Train Data
Test Data
Tweets Preprocessing
Corpus Based Features Extraction
Random Forest Tree – Binary Classifier
Lexicon Based Feature Extraction
Conditional Random Fields
Tagged Output Sequence

Figure 1: Model Diagram of Proposed Approach

Entity	Tamil-English	Hindi-English
ARTIFACT	18	25
LIVTHINGS	16	7
DISEASE	5	7
COUNT	94	132
DATE	14	33
FACILITIES	23	10
PERSON	661	712
DISTANCE	4	-
SDAY	6	23
MONTH	25	10
DAY	15	67
PLANTS	3	1
MATERIALS	28	24
TIME	18	22
MONEY	66	25
ENTERTAINMENT	260	810
LOCATION	188	194
LOCOMOTIVE	5	13
ORGANIZATION	68	109
PERIOD	53	44
YEAR	54	143
QUANTITY	-	2

Table 3: Entity Tags Statistics

The overall approach is performed in a system with following specification: Linux operating system, python3.4, 16 GB RAM and 8 core processor. In order to perform CRF, Sklearn - CRFSuite[3] is utilized, TDM matrix is built using sklearn-CountVectorizer[4] library, Random Forest Tree classifier[5] is from sklearn library, part of speech tagging done using NLTK[6] library and preprocessor using twitter-preprocessor[7].

The statistics about both the data-sets are given in Table 2 and Table 3. Initially raw tweets are tagged with its corresponding entities given in Table 3 with respect to the annotation file provided by Code Mixed Entity Extraction -Indian Language task organizers.

Since the given data-set is tweet, the tendency of noise presence is higher and unwanted text, non-text information will lead to build a sequential model with low performance. These unwanted informations, web links and emoticons are removed from tweets through twitter preprocessor.

Followed by the preprocessing step, a set of corpus based features are extracted out of the entity words in a tweet to built the Random Forest Tree based binary classifier. For extraction initially all the entities in the training corpus are re-tagged as 'Entity' and others as 'not a Entity'. From the entity words present in the training corpus their corresponding prefix-suffix of length 1 to 4 are taken to build the vocabulary for TDM matrix by using CountVectorizer.

The TDM matrix is built based upon the presence of prefix, suffix information present within the words. Along with this TDM matrix length of the word, position of the word lies in its tweet and total number of times the prefix or suffix present in the corpus are taken as attributes to train the Random Forest Tree classifier. $NC_{\sqrt{N}}$ number of trees are utilized to built the Random Forest tree, where N is the total number of attributes. Similarly testing corpus is also applied on the above steps to get the given word is entity or not. In order to measure the training performance 10-fold 10-cross validation is carried out and obtained near 96%, 97% respectively for the Tamil - English and Hindi - English corpus.

With the above obtained binary feature, other features mentioned in the Table 1 are extracted out of the training corpus. A window of length 5 is taken to capture the context of word as well as features by taking previous two words and later two words from the current word. Using these features as the constraint function CRF sequential model is built for entity recognition task. Similarly features are extracted for testing and output labels are predicted for input testing word sequences. Finally words with the consecutive output labels are concatenated together to form phrases with single tag. To ensure the training performance, similar to Random Forest Tree here also cross validation is carried over and obtained nearly 94% as the precision for both the corpus.

The performance against the test set of top 5 teams are given in Table 4 and Table 5. It can be observed that from the top score the precision of the proposed system only varies around 2% in Hindi - English corpus and almost equal in Tamil - English Corpus. The problem arises with the recall, which affects final F measure. Hence our future work will be focused on improving the recall of the proposed system.

5. CONCLUSION

Conditional Random Field based Entity Recognition with hybrid features was experimented on CMEE - IL (Code Mixed Entity Extraction - Indian Language) corpus and attained greater performance. The experimented approach

[3] pypi.python.org/pypi/sklearn-crfsuite
[4] scikit-learn.org
[5] scikit-learn.org
[6] www.nltk.org
[7] github.com/s/preprocessor

Team	Precision	Recall	F
Deepak-IIT-Patna	79.92	30.47	44.12
Veena-Amritha-T1	79.51	21.88	34.32
Bharathi-Amrita-T2	**79.56**	**19.59**	**31.44**
Rupal-BITS-Pilani-R2	58.71	12.21	20.22
Shivkaran-Amritha-T3	47.62	13.42	20.94

Table 4: Results : Tamil - English

Team	Precision	Recall	F
Irshad-IIIT-Hyd	80.92	59.00	68.24
Deepak-IIT-Patna	81.15	50.39	62.17
Veena-Amritha-T1	79.88	41.37	54.51
Bharathi-Amrita-T2	**77.72**	**31.84**	**45.17**
Rupal-BITS-Pilani	58.84	35.32	44.14

Table 5: Results : Hindi - English

performed great on both the Tamil-English and Hindi-English tweet corpus by attaining nearly 95% against the training corpus and 45.17%, 31.44% against the testing corpus. Preprocessing of social media text is an essential part. This will improve the feature engineering (reduces the sparsity) and boost the performance of the proposed system. Hence the future work will be focused on incorporating necessary pre-processing steps along with the proposed approach.

6. REFERENCES

[1] N. Abinaya, N. John, H. B. Barathi Ganesh, M. Anand Kumar, and K. Soman. Amrita_cen fire-2014: Named entity recognition for indian languages using rich features. pages 103 – 111, December 2014.

[2] H. B. Barathi Ganesh, N. Abinaya, M. Anand Kumar, R. Vinayakumar, and K. Soman. Amrita-cen neel: Identification and linking of twitter entities. 2015.

[3] U. Barman, A. Das, J. Wagner, and J. Foster. Code mixing: A challenge for language identification in the language of social media. volume 13, 2014.

[4] L. Breiman. Random forests. volume 1, pages 5–32, October 2001.

[5] A. Das and B. Gamback. Code-mixing in social media text.

[6] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. volume 1, pages 282–289, June 2001.

[7] D. Maynard, K. Bontcheva, and D. Rout. Challenges in developing opinion mining tools for social media. pages 15–22, 2012.

[8] J. Piskorski and R. Yangarber. Information extraction: Past, present and future. 2013.

[9] A. PVS and G. Karthik. Part-of-speech tagging and chunking using conditional random fields and transformation based learning. volume 21, 2007.

[10] A. Ritter, S. Clark, and O. Etzioni. Named entity recognition in tweets: an experimental study. pages 1524–1534, July 2011.

[11] J. Tang, M. Hong, D. Zhang, L. B, and L. J. Information extraction: Methodologies and applications. October 2007.

[12] Y. Vyas, S. Gella, J. Sharma, K. Bali, and M. Choudhury. Pos tagging of english-hindi code-mixed social media content. volume 14, pages 974–979, October 2014.

[13] D. Westerman, P. Spence, and B. Van Der Heide. Social media as information source: Recency of updates and credibility of information. volume 19, pages 171–183, January 2014.