Journal of Computer Science

Original Research Paper

Named Entity Recognition for Kannada using Gazetteers list with Conditional Random Fields

[1]K.P. Pallavi, [2]L. Sobha and [3]M.M. Ramya

[1]Department of Computing Sciences, Hindustan Institute of Technology and Science, Chennai, India
[2]AUKBC, MIT campus, Chennai, India
[3]Center of Automation and Robotics, Hindustan Institute of Technology and Science, Chennai, India

Article history
Received: 03-10-2017
Revised: 15-02-2018
Accepted: 23-02-2018

Corresponding Author:
K.P. Pallavi,
Department of Computing Sciences, Hindustan Institute of Technology and Science,
Chennai, India
Email: pallavi.abhijith@live.in

Abstract: Named Entities (NEs) that exist in the sentences are essential to build Natural Language Processing (NLP) applications for Information Extraction (IE) from large corpora. However, generating a large corpus is challenging for resource poor languages, such as Kannada. Further, there is no annotated corpus available online. The challenges faced in annotating NEs with pre-defined classes are: It is morphologically joined with other words and the spelling variations are more frequent for Kannada words. Sentence structure varies according to morphology, parts of speech (pos) and chunking of a language. These parameters differ from one language to another. To address these challenges, a novel application system is proposed to identify NEs in Kannada using a large corpus of 73,676 tokens. The Named Entity Recognition (NER) system consist of a robust pos tagger and Noun Phrase (NP) chunker developed for generic data. Five gazetteer lists were created from many orthographic patterns for each word. Context information such as previous two words, next two words, word morphology and gazetteer lists were added to feature lists. An unigram-bigram template was designed and incorporated into Conditional Random Fields (CRFs) to generate conditional feature functions. The proposed system resulted in 86.85% and 71.01% f-measure for gold test data and newspaper data respectively.

Keywords: Named Entities, Natural Language Processing, Noun Phrase Chunker, Conditional Random Fields

Introduction

Kannada is the official language of the state of Karnataka, a major state in south India with a population of 64 million; it has 70 million speakers (Bhat, 2012) all over the world. It is a language with rich morphology and agglutination like other Indian languages. Morphology is the study of words where the smallest grammatical units change the word meaning. Nouns are added with suffixes to form meaningful words and sentences, whereas Kannada nouns are more agglutinate with suffixes compared to other Indian languages (Bhat, 2012). Nouns are marked with case, verb and png (person, number, gender) markers which make it very hard to identify the root nouns. Many times, more than one word joined in a sentence, but the meaning remains the same for individual and together representation. Those kinds of blended words are tough to separate orthographically as well as morphologically. Separating negation words from the nouns is more challenging. Moreover, a single word in Kannada can be written in many orthographic forms. Orthography of language varies according to the software used to type and it depends on the writer. An online resource for Kannada language is inadequate due to the above causes. Currently there are no freely available large online corpus and gazetteer lists for Kannada. There are only few NERs, pos taggers and chunk taggers reported (Amarappa and Sathyanarayana, 2013a; 2013; 2015; Bhuvaneshwari, 2014; Pallavi and Pillai, 2015).

Today, there is immense interest in NLP due to the advances in technology. NLP is one of the pioneering aspects of human-machine communication, where the language is one of the most basic needs of human communication. The prime objective of NLP is to develop models to process linguistic tasks like reading, writing, hearing and speaking (James, 1995). NER is a central task of NLP. It requires identification of proper

Science Publications

© 2018 K.P. Pallavi, L. Sobha and M.M. Ramya. This open access article is distributed under a Creative Commons Attribution (CC-BY) 3.0 license.

names from unstructured data and classifying them into a set of predefined categories of interest. Categories like the names of people, organization names, date, time, etc., are grouped under three classes in the first level: Entity Names (ENAMEX), Numerical (NUMEX) and Time expressions (TIMEX). The different properties of these classes make it easier to classify into further subclasses (Nadeau and Sekine, 2007; Malarkodi *et al*., 2012). A similar approach was also used in this proposed system.

The proposed Kannada NER model in this study has been trained based on the pos tags, NP chunk tags, context of a sentence, affixes of words and gazetteer lists. NER model was built using a supervised Machine Learning technique called CRFs. It was introduced by Lafferty *et al*. (2001) to build a probabilistic model to segment and label the sequence data. It can be applied to text and speech processing, including topic segmentation, pos tagging, information extraction and syntactic disambiguation. The advantages of CRFs are: It can solve large dependency problems and label bias problems. Hence the author used CRF for the proposed system.

The study is presented as follows: Section two carries out a survey on NER. Section three describes CRFs and section four explains the architecture for the Kannada NER system. In the end, the error analysis and experimental results are discussed in section five and concluded in section six respectively.

Related Work

An extensive literature survey was performed on language independent models. A first language independent model for NER was proposed (Cucerzan and Yarowsky, 1999) using an un-annotated text to learn bootstrapping algorithm, then trained on a very small labeled named list. Identified NEs with the help of morphological and contextual information extracted using hierarchical trie models. It gave the accuracy between 70 to 79% for 5 languages. A language independent NER using a maximum entropy was developed (Curran and Clark, 2003) to identify locations, organizations, persons and miscellaneous NEs for English, Dutch and German languages. The NEs feature set includes POS and chunk information, period, punctuation and numbers with annotated data. Data sets were collected from the CoNLL shared task, where English reached better accuracy of 84.89% compared to German of 68.41% and Dutch by 79.61%. Another paper was reported on Language-Independent Named Entity Recognition (Tjong Kim Sang and De Meulder, 2003). It was the evaluation report on the performance of sixteen NER systems from the CoNLL shared task. The evaluation system has been trained, developed and tested for English and German data sets. The best Identified systems were English and German. Later, the results of those systems were improved by reducing the error rate of 14% for English and 6% for German. One more paper (Nothman *et al*., 2013) experimented with nine European languages by evaluating English, German, Spanish, Dutch and Russian data sets. Millions of words from Wikipedia were annotated to train NER model. That resulted with an accuracy of 94.9% for a person, location and organization names and 89.9% for fine grained entity types.

An un-annotated text to learn bootstrapping algorithm was proposed with a very small labeled name list. NEs were identified with the help of morphological and contextual information extracted from hierarchical data and structural model. Such independent NERs were developed for various European languages (Nadeau and Sekine, 2007). Recently there are many NERs proposed for Asian languages like Malay (Murthy *et al*., 2016; Noor *et al*., 2016; Sulaiman *et al*., 2017).

These systems, however, cannot identify NEs from Indian languages due to its complexity in terms of morphology and agglutination. One NE may give many meanings which leads to confusion when classifying it. Indian names are based on variety of conventions like epic names, celestial bodies and so on. Nested Entities are more usual in Indian names.

The above challenges have been overcome with the existing NERs using both linguistic features and statistic methods to develop NER (Pandian *et al*., 2007). They began with preprocessing, then extracted the clues from the words with the morphological analyzer. The words underwent through semantic and shallow parsers and learned the system using statistical processing to identify NE's. Identified NEs were used to generate an automatic dictionary. The accuracy achieved was 72.72% In another paper (Ekbal *et al*., 2008), NER with 17 tags were identified from partially tagged corpus using features like context word feature, word suffix, word prefix along with Gazetteers lists. An achieved f-measure was 90.7% using CRF.

Few more Indian Language NER systems were developed for Hindi (Saha *et al*., 2008a; 2008b; Nayan *et al*., 2008), Bengali (Ekbal *et al*., 2008; Ekbal and Bandyopadhyay, 2008), Tamil (Pandian *et al*., 2007; Vijayakrishna and Devi, 2008; Malarkodi *et al*., 2012), Telugu (Sulaiman *et al*., 2017; Shishtla *et al*., 2008a; 2008b; Srikanth *et al*., 2008), Malyalam, Marati, Punjabi, Oriya (Gali *et al*., 2008) and Urdu (Riaz, 2010) using supervised machine learning techniques (Wei, 2004). The techniques used were CRFs, Maximum Entropy, Hidden Markov Model (HMM), Support Vector Machines (SVM) and Maximum Entropy Markov Models (MEMM). They were also combined with rule based to form a hybrid approach.

The NERs for Kannada were proposed using HMM, Navie Bayes and rule based (Amarappa and Sathyanarayana, 2013a; 2013b; Bhuvaneshwari, 2014). A Rule based system was implemented with 16 contextual rules with 8 features. Additionally, some significant information such as pos and chunk information provides useful linguistic properties, which

were considered to identify NEs in the current work. Since CRFs perform better than HMM and MEMM in terms of dependency problems and label bias problems, it has been widely used to generate NER models. Hence it is suggested to use CRFs for Kannada NER.

CRF

CRFs generates probabilistic model for the given sequence data. Sentence based prediction depends on the sequence of inputs given $x = \{x_1, x_2, x_3, ..., x_T\}$. Features information like word, word morphology, pos tag of each word in a sentence and word prefix and suffix units were saved in x, where the output pattern classes were stored in $y = \{y_1, y_2, y_3, ..., y_T\}$. Each element of vector y is called a tag. Tags were the labels given for each word. Predicted tag y of each word might be a sequence in probability of x and y which is generated from a set of feature functions $f = \{f_1, f_2, ..., f_T\}$. For example, the current word w_i, the sentence s, i^{th} position, previous word w_{i-1} computes a feature function $f(s, i, w_i, w_{i-1})$. Here current word w_i depends on the previous word w_{i-1}. Generating model using joint probability is difficult due to the complex dependencies among the largest feature sets. But, it can be done using conditional probability. Prediction takes the advantage from joint probability where conditional probability supports classification. The Conditional probability for a linear chain CRF is as follows (Gali *et al*., 2008):

$$P(y_{1:T}|x_{1:T}) = \propto x^{[\sum_{i=1}^{T} F(y_{i-1}, y_i, x, i)]}$$

where, α is the normalization factor and $F(y_{i-1}, y_i, x, i)$ is the sum of feature functions.

Architecture of Kannada NER System

The architecture of a Kannada NER system is shown in Fig. 1. The raw data was downloaded from Kannada Wikipedia. The unnecessary information like pictures and labels were manually removed before preprocessing. Then the data was given to the pre-processor, tokenizer and annotator. Features were extracted from the data and it was randomized and 5-folded for validation.

Finally, CRF models were used to identify and classify NEs. The classified NEs were saved in tagged data. The few classes in the input file were modified based on the error analysis which is explained in Annotation section. The process was repeated until the system reached precision above 95% for the training set. Each module in the proposed system is described below.

Pre-Processing and Tokenization

Three pre-processing rules were designed and applied on collected and cleaned data. The first rule was applied to separate the symbols and punctuation marks from the words which occur together in the corpus. It helped in reducing preprocessing time. In the examples given in Table 1, symbols are separated from the word with a blank space. Similarly, the second rule splits the orthographically joined words, which helped in NE identification. The third rule separated the morphological blended nouns from all kinds of verbs such as finite verb, infinite verb and so on. For example, the nouns were separated from pronouns and verbs without changing the meaning of the sentence, as shown in Table 2. This led to better classification of the NEs.

Python tokenizer was written to split the pre-processed sentences into words and emblematized in columns. A blank space was the separator for the tokenizer. Also, different spellings of each gazetteer word have been included into the list. Developing gazetteer lists imposed many difficulties, as Kannada characters were represented using two different unicode's. For example: '◌ೈ' is written as a single character and the unicode of that is '\u0cc7', whereas it is also represented by two characters '◌ೈ' and '◌ೆ' and unicode of those two were '\u0cc6' and '\u0cd5'. Pre-processing rules such as, '*If the unicode '\u0cd5' present in the string, the previous character should increment by 1 and current unicode should replace it with NULL'* helped to overcome these challenges.

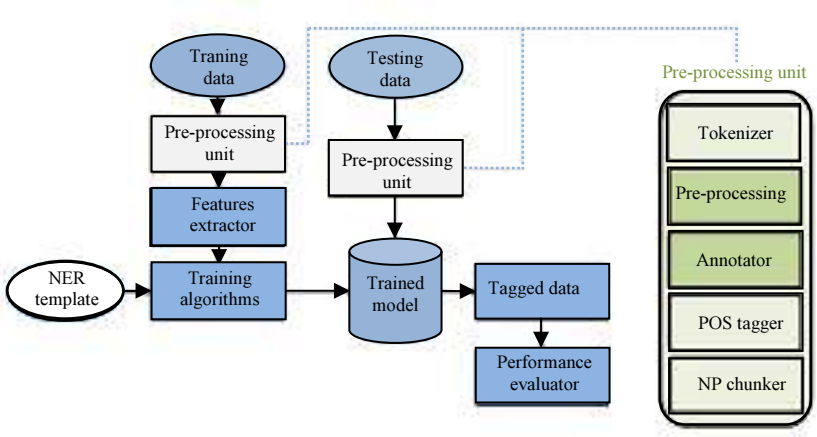

Fig. 1: Architecture for kannada NER system

Table 1: The words given before and after pre-processing

Before pre-processing	After pre-processing
೧೯೩೪-೧೯೬೪ (1934-1964)	೧೯೩೪ – ೧೯೬೪(1934 – 1964)
'ಪ್ರಗತಿ' (‘pragathi’)	‘ ಪ್ರಗತಿ ’ (‘ pragath ‘)
ಬೆಂಗಳೂರು, (bengaluru,)	ಬೆಂಗಳೂರು , (bengaluru ,)
ಅ.ನಾ.ಪ್ರಹ್ಲಾದರಾವ್ (a.naa.prahlaadharaav)	ಅ . ನಾ . ಪ್ರಹ್ಲಾದರಾವ್ (a . naa . prahlaadharaav)

Table 2: Fused Nouns before and after separation

Fused nouns	Nouns after separation
ರಾವ್ರವರು (raavravaru)	ರಾವ್ ಅವರು (raav avaru)
ಅಟ್ಲಾಂಟಿಕ್ ರಿಡ್ಜ್ನಲ್ಲಿ (atlanticrigdenalli)	ಅಟ್ಲಾಂಟಿಕ್ ರಿಡ್ಜ್ ನಲ್ಲಿ (atlanticridge nalli)
ಭಾರತದಲ್ಲಿರುವ (bharathadhalliruva)	ಭಾರತದಲ್ಲಿ ಇರುವ (bharathadhalli iruva)
ಕಲಾಂರವರೆಂದರೆ	ಕಲಾಂರವರು ಎಂದರೆ (kalaamravaru endhare)
ಭಾವನಗರ್ನಲ್ಲಿರುವ (bhavanagrnalliruva)	ಭಾವನಗರ್ ಅಲ್ಲಿರುವ (bhavanagr alliruva)

Tagset

Tagset is a collection of tags. The hierarchical standard tagset consists of 106 tags totally referred by Malarkodi *et al*. (2012) used to develop Kannada NER system. It is divided into three main classes ENAMEX, TIMEX and NUMEX. Further, ENAMEX is subdivided into 22 classes, NUMEX into 4 classes and TIMEX into 7 classes. Tags used are shown in Table 3. Labeling was done using the BIO format as shown in Table 4. ‘B’ symbolized beginning of the entity, ‘I’ substantiated to the inside entity and the nonentities were marked with ‘0’.

Annotation

Annotating large corpus is a time consuming task. A base model was trained using manually annotated 6000 words. The remaining corpus was tested using the base model. Then it was verified manually by the author and added to the existing base engine to train again. The process was repeated to tag complete corpus. Boundaries of named entities were considered while tagging and the example of annotated data is given in Table 4.

Here, INDIVIDUAL is a label of NE which represents the individual person’s name. The types and occurrences of NEs vary from article to article in generic corpus. That makes it difficult to recognize NEs which are common nouns. Some common nouns like “ವಿಮಾನ” (flight) and “ಗಗನ ನೌಕೆ” (space shuttle) were tagged automatically as locomotives by the base engine. The non NE’s were untagged manually. For example, “ತಂದೆ” (Father), “ತಾಯಿ” (Mother) and “ಭಾಷೆ”(language) which are common nouns and not NEs. Few other labels like “ಇಂಗ್ಲೆಂಡ್” (England), “ಗ್ರೇಟ್ ಬ್ರಿಟನ್” (great britan) were tagged as regions initially, later these were changed into Nations.

Features used for Kannada NER

POS

POS of each language differ from others, it is very difficult to find the sentence structure in Indian languages due to its free word order nature. POS tagger using CRFs has been developed. There are three noun categories (common noun, proper noun, location) in the pos tagset (Pallavi and Pillai, 2015) which helped to identify named entities. Often, NE is always represented as a proper noun. POS tagger was trained on 64K tokens which were collected from Kannada Wikipedia. It was tested on 16k tokens. NEs were also identified with the support of other features like noun phrase chunk tags.

Noun Phrase Chunk Tags

The process of identifying and labeling phrases in a sentence is known as Chunking. It classifies the phrases into Noun Phrase (NP), Verb Phrase, Adjectival Phrase etc (Pattabhi *et al*., 2007). This is the basic element for all NLP applications including NER. NEs are all nouns and it occurs mostly in NP. Hence, only Noun phrase chunker considered in this study. Features such as words, pos and NE tags of each word were trained with the help of CRFs to develop a Kannada NP chunker.

Affixes

These are derivation inflections of a word. Affixes used in the NER system are:

- Suffix: Occurs at the end of the word and the last character of each word was used as a suffix feature to group the NEs which ends with similar Kannada alphabets
- Prefix: Occurs at starting point of words. Sometimes prefix pattern matches between NEs like ‘ರ’ (‘ra’) alphabet is same in ‘ರತ್ನ ದೀಪ’, ‘ರುಕ್ಮಿಣಿ’, ‘ರೂಪಶ್ರೀ’ (‘rathnadeepa’, ‘rukmini’, ‘roopashree’) that supports classification. Starting with three characters of each word considered as a prefix feature
- Case markers (Vibakthi): A grammatical form of words which don’t have any particular lexical meaning, but it performs grammatical functions. Vibakthis are the case markers in Kannada which occur with a noun as a suffix and they are eight in number. This assisted in identifying the nouns from the corpus

Gazetteer Lists

Creating a gazetteer list is difficult for Kannada due to unicode variations of the script. Gazetteer lists without unicode variations cannot give better accuracy. Hence, spelling and unicode variations of the gazetteer lists were considered. Totally 5 gazetteer lists were used in this study: Days of a week, Months and location names. Location gazetteer list consist of only countries and state names.

Table 3: Tags used in the current work

Enamex	City	Continental	Institutes
Individual	District	Medicines	Organisms
Family name	State	Artifacts	Non-humans
Title	Nation	Chemicals	Celestial
Group	Continent	Entertainment	Bodies
Organization	Address	Cinema	Distance
Government	Street	Drama	Money
Private Company	Water-Bodies	Sports	Quantity
Public company	Rivers	Events	Count
Religious	Religious places	Conferences	Time
Non-profit organization	Materials	Others	Year
Charitable	Locomotives	Disease	Month
Association	Metals	Treatment	Date
GPE (Geo-political social entity)	Fruits	Museum	Day
Media	Plants	Parks	Period
Location	Cuisines	Monuments	Festivals
			Facilities

Table 4: Examples for tagged words

Words	Tags
ಅರಕಲಗೂಡು(Arkalaguudu)	B-INDIVIDUAL
ನರಸಿಂಗರಾಯ(Narasingaraya)	I-INDIVIDUAL
ಕೃಷ್ಣರಾಯ (Krishnaraya)	I-INDIVIDUAL
ಕನ್ನಡ (Kannada)	B-ENAMEX
ಸಾಹಿತ್ಯಲೋಕದ(saahithyalookadha)	0
ಪ್ರಮುಖರಲ್ಲೊಬ್ಬರು(pramukharalobbaru)	0
.	0

Table 5: Corpus statistics

Corpus	Training	Testing	Randomized training	Randomized testing
Number of tokens	58938	14737	58939	14730
Number of NEs	6533	1692	6563	1662
OOV of NEs	419	1273	871	791

Corpus

Articles on various areas such as sports, natural calamities and pilgrimages were collected from Kannada Wikipedia as raw data. The annotated corpus was generated with 73,676 words after pre-process. Articles are different from each other and there are many different types of NEs present. This Corpus was divided into 80:20 ratio for training and testing, respectively. There are unique words present in test data compared to training data and those are called as Overall Out of Vocabulary (OOV) words. OOV words in the test data were 40%, which increased the complexity of developing a robust NER system. More than 75 and 40% of NEs are OOV for training and randomized training set respectively. Exact numbers are given in Table 5.

Another new dataset was created using 2K words for final evaluation. That was collected from Kannada daily newspaper called ‘Vijaya Karnataka’(Kannada daily newspaper).

Randomization and 5-fold Validation

NE identification and classification was ontology independent for each sentence. The context of a sentence was not depending on previous or next sentence. Hence, the sentences in the corpus were arranged in a random order and it was considered for experiments. Corpus is divided into 5 folds. Each time one and four folds were used for testing and training respectively. Collectively 5 sets of experiments were conducted to examine the NER model on different sentence sequences.

Template for Kannada NER

A probabilistic NER model was generated for structured sentences and sequence of their dependencies framed using feature functions (features might be words, pos tags, etc.,). Feature selection was the primary and predominant task which influenced the system towards attaining good accuracy. Those features were utilized to design a template by analyzing input data. The template

was the key factor for CRFs. Unigram and bigram feature template, weights of those feature functions and normalization factors of conditional probability distribution propagated a NER model. All the individual features represented in unigram and the combination of features represented bigram. The test data passed through model for identification and classification of NEs. It has been saved in a text file and performance evaluation was done to find the results.

Performance Evaluation

Target labels of the NER system considered the boundaries in BIO format and labels of NEs. The performance of Kannada NER was measured in terms of precision, recall, f-measure:

$$Precision = (true\ positives) / (true\ positives + false\ positives)$$
$$Recall = (true\ positives) / (true\ positives + false\ negatives)$$
$$F - measure = (2 * Precision * Recall) / (Precision + Recall$$

Where:

- True positives represent the number of NE's classified correctly
- False positives represent the number of NE's classified for non NEs
- False negatives represent the number of NE's not classified for correct NEs

Results and Discussion

The performance of any NER system depends upon the data used for training. Generic data include all types of NEs which can be used in any application. Hence, the publicly available online articles from Kannada Wikipedia were collected. The corpus was cleaned, preprocessed and annotated. The annotation process of the corpus increased system performance according to the experimental observations. The primary part of the annotation is a classification of NEs and it led to confusion in many cases like.

ಕಾಶಿ (Kashi) This belong to two classes city name and religious places. It is tagged depending on the context of a sentence.

ಸ್ಕಾರ್ಬರೋ ಪಟ್ಟಣದಲ್ಲಿ (In Scarborough city) Here the 'Scarborough' is a city name which belongs to the city class and it is the beginning of the entity. The 'in city' word was tagged as an inside entity because it was appended with 'Scarborough'. In a few cases 'in city' appears independently without nesting to any NE, those were not tagged with any labels.

40**ಕ್ಕೂ ಹೆಚ್ಚು** (more than 40) Similarly, this considered as a single NE and classified as count class. '**ಕಂಚಿನ ಯುಗದ**' (bronze century), '**ರೋಮನ್ ಅವಧಿಯಲ್ಲಿ**' (In Roman period), '**ಹಲವಾರು ವರ್ಷಗಳ**' (For many years) NEs belonged to period class. The independent words '**ಹೆಚ್ಚು**'(more), '**ಯುಗದ**' (century), '**ಅವಧಿಯಲ್ಲಿ**' (period), '**ವರ್ಷಗಳ**' (years) were not tagged because they were not joined with any NE.

Annotated tokens were passed through CRFs along with selected features. Features were basic words, pos tags, noun chunk tags and gazetteer list in the first phase as shown in Table 6.

The pos tagger developed by the author achieved an accuracy of 92.8% (Pallavi and Pillai, 2015) and the new NP chunker with 95.32% accuracy was developed for the task were used to improve Kannada NER system. The system was trained using pos and chunk information in the beginning. Later, gazetteer lists with 7 continents were listed and were matched with the corpus. The system could only match only 5 gazetteer out of 59 from the corpus, due to different kinds of spellings and unicodes. This was solved by adding unicode rules to pre-processing, then the gazetteer lists helped to increase the accuracy by 0.72%.

In the second phase: vibakthi, prefix and suffix were included for training. The system attained competitive f-measure for 80:20 ratio of a corpus and it is shown in Table 7.

In the third phase, randomization experiment was carried out and it shows that f-measure increased by 4.48% from p3 matched to p2. Number of NEs present in the randomized training corpus, compared to non-randomized training corpus was more. It helps the system to easily select the similar statistical functions to classify NEs. For example, **ಪಂಜಾಬ್** (paNjAb) occurred only once in non-random training corpus and it occurred 5 times in random training corpus. System trained with random data was able to classify **ಪಂಜಾಬ್** (paNjAb) as a state correctly, whereas, system trained without random data was unable to classify.

Random data contain the additional number of NE classes. Hence, the time needed to train n-folds of p3 are higher than p2 which is shown in Fig. 2. Training time increased around 14 minutes for randomization, but the testing time do not result in much difference. The 5-fold experiment has been conducted on both randomized and unrandomized data. The mean was calculated for both and was found to be approximately equal as shown in Table 7. N-fold validation results were improved compared to corpus results(p2). Table 7 shows that the 5-folds attained almost same f-measures for p3 and n-fold randomization of p3. Hence, it proves that the occurrences of NEs of each class in the training data are necessary to accomplish a high f-measure (Wang and Patrick, 2009).

Table 6: List of Features used

Phase	Features used
p1	pos tags, chunk tags, gazetteer lists
p2	pos tags, chunk tags, vibakthi, prefix, suffix, gazetteer lists
p3	Sentence randomization of overall corpus (p2)

Table 7: Precision, recall and f-measure of Kannada NER

Evaluation matrix	Using p2	Using p3	Mean of 5-fold using p2	Mean of 5-fold using p3
Precision	97.12	95.21	96.38	96.23
Recall	78.55	87.75	87.84	87.84
f-measure	86.85	91.33	91.91	91.84

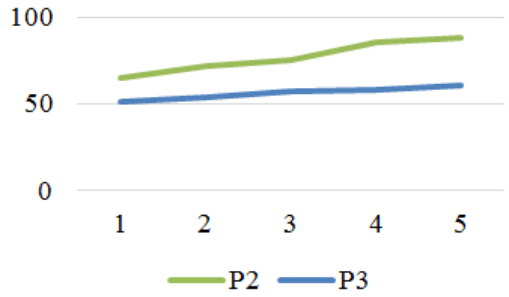

Fig. 2: 5-fold training

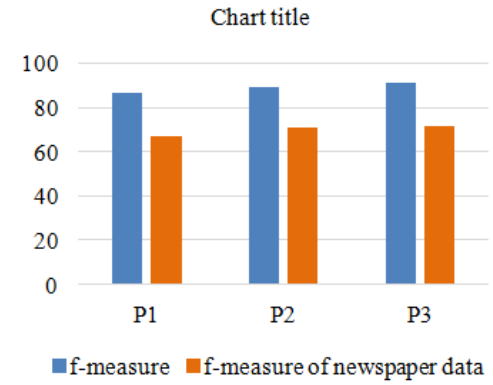

Fig. 3: F-measure of gold test sets and newspaper data sets

Sampling test was conducted on the same corpus. Similar NE's that occur more than 2 times in the training set have been removed. It took 31 min to train which is 8 mins less than randomization.

Newswire data with 13232 tokens were tested. Fig. 3 shows both newswire and gold test corpus results. Newswire data set f-measure is reduced due to lack training information. Like, various combinations of features helped to attain good f-measure of gold test set, but that was not sufficient to identify NE's in newswire. For example, the number of NE classes present in the corpus is 247 for nation and 68 months. f-measures of both are 86 and 92%, which are higher compared to other classes, due to the use of gazetteer lists in the system.

Error Analysis

Analysis was conducted subjectively for all the classes, based on the results obtained. The major issue observed was that the nested NE's don't work with this method. For example, PERIOD class consist of date, month, year and special symbol which makes the system to tag classes individually rather than one entity. Similarly, nested NEs were existed in ASSOCIATION, EVENT and ORGANIZATION classes.

Types of NEs differ from article to article and those make it hard for system to tag them correctly, unless an excellent generalized POS tagger is available. POS is the key element for NE classification and an automated POS tagger used in this system was with a 7.8% error rate. Some incorrect POS tags like proper nouns tagged as common nouns are misguiding the NE classifier as well.

Not much information was available for the NE classes such as events, media and association. The suffixes of those were matched with the other NEs like a person and location. It was difficult to handle these kind of NEs in Indian languages.

Conclusion

The NER system automatically identified the named entities from the Kannada corpus and it also classified them into different categories. Some of the main categories like individual person names, country names, continent names, CRFs state names, date, month names, count, government organization names, group names gave competitive results. The results improved after using pos tags, chunk tags, noun case markers, suffixes and prefixes of the words. Along with that, unigram, bigram and contextual information's helped to attain better accuracy. All this information was used to generate CRF model, which identified the Named Entities in the given data. The Proposed system achieved an f-measure of 91.33% for randomized sentences (p3) with an increase in time compared to normal corpus (p2). Feature p2 processed better for online training where p3 processed better for offline training.

Acknowledgement

The authors would like to thank Hindustan Institute of Technology and Science for their support. The authors wish to present special thanks to Language Technologies and Knowledge Discovery Group, AU-KBC research center, Chennai for their constant support and encouragement.

Author's Contributions

K.P. Pallavi: Conducted experiments and written the article.

L. Sobha: Contribution towards designing the experiments and acquisition of data.

M.M. Ramya: Drafting and reviewing the article.

Ethics

The authors have not used any one else's database. The corpus was created by the author using freely available online resource that is Kannada Wikipedia.

References

Amarappa, S. and S.V. Sathyanarayana, 2013. Named entity recognition and classification in Kannada language. Int. J. Electron. Comput. Sci. Eng., 2: 281-289.

Amarappa, S. and S.V. Sathyanarayana, 2013. A hybrid approach for Named Entity Recognition, Classification and Extraction (NERCE) in kannada documents. Proc. Int. Conf. Multimedia Process. Commun. Info. Tech.

Amarappa, S. and S.V. Sathyanarayana, 2015. Kannada named entity recognition and classification (nerc) based on multinomial naïve bayes (mnb) classifier. Int. J. Natural Language Comput. DOI: 10.5121/ijnlc.2015.4404

Bhat, S., 2012. Morpheme segmentation for Kannada standing on the shoulder of giants. Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing, (NLP' 12), pp: 79-94.

Bhuvaneshwari, C.M., 2014. Rule based methodology for recognition of kannada named entities. Int. J. Latest Trends Eng. Technol., 3: 50-59.

Cucerzan, S. and D. Yarowsky, 1999. Language independent named entity recognition combining morphological and contextual evidence. Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, (VLC' 99), pp: 90-99.

Curran, J.R. and S. Clark, 2003. Language independent NER using a maximum entropy tagger. Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL, (LLH' 03), Edmonton, pp: 164-167. DOI: 10.3115/1119176.1119200

Ekbal, A., R. Haque and S. Bandyopadhyay, 2008. Named entity recognition in Bengali: A conditional random field approach. IJCNLP.

Ekbal, A. and S. Bandyopadhyay, 2008. Bengali named entity recognition using support vector machine. IJCNLP.

Gali, K., H. Surana, A. Vaidya, P. Shishtla and D.M. Sharma, 2008. Aggregating machine learning and rule based heuristics for named entity recognition. IJCNLP.

James, H., 1995. Natural Language Understanding. 1st Edn., Dorling Kindersley pvt. Ltd., New Delhi, India.

Lafferty, J., A. McCallum and F.C. Pereira, 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Malarkodi, C.S., R.K. Pattabhi and L.D. Sobha, 2012. Tamil NER - coping with real time challenges. Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages, (PIL' 12), pp: 23-23.

Murthy, V., M. Khapra and P. Bhattacharyya, 2016. Sharing network parameters for crosslingual named entity recognition. Comput. Sci.

Nadeau, D. and S. Sekine, 2007. A survey of named entity recognition and classification. Lingvisticae Investigationes, 30: 3-26.

Nayan, A., B.R.K. Rao, P. Singh, S. Sanyal and R. Sanyal, 2008. Named entity recognition for indian languages. Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, (EAL' 08), pp: 97-104.

Noor, N.M., J. Sulaiman and S.A. Noah, 2016. Malay name entity recognition using limited resources. Adv. Sci. Lett., 22: 2968-2971. DOI: 10.1166/asl.2016.7124

Nothman, J., N. Ringland, W. Radford, T. Murphy and J.R. Curran, 2013. Learning multilingual named entity recognition from Wikipedia. Artificial Intell., 194: 151-175. DOI: 10.1016/j.artint.2012.03.006

Pallavi, K.P. and A.S. Pillai, 2015. Kannpos-kannada parts of speech tagger using conditional random fields. Proceedings of the Emerging Research in Computing, Information, Communication and Applications, (ICA' 15), Springer India, pp: 479-491.

Pandian, S., K.A. Pavithra and T. Geetha, 2007. Hybrid three-stage named entity recognizer for tamil. INFOS.

Pattabhi, R.K., T. Rao, S.R.R.R. Vijay, Vijayakrishna and L. Sobha, 2007. A text chunker and hybrid POS tagger for Indian languages. Shallow Parsing South Asian Languages.

Riaz, K., 2010. Rule-based named entity recognition in Urdu. Proceedings of the 2010 Named Entities Workshop, Jul. 16-16, Uppsala, pp: 126-135.

Saha, S.K., S. Sarkar and P. Mitra, 2008a. A hybrid feature set based maximum entropy hindi named entity recognition. IJCNLP.

Saha, S.K., S. Chatterji, S. Dandapat, S. Sarkar and P. Mitra, 2008b. A hybrid approach for named entity recognition in Indian languages. Proc. IJCNLP.

Shishtla, P., P. Pingali and V. Varma, 2008a. A character n-gram based approach for improved recall in Indian language NER. IJCNLP.

Shishtla, P., K. Gali, P. Pingali and V. Varma, 2008b. Experiments in Telugu NER: A conditional random field approach. IJCNLP, (2008, January) pp: 105-110.

Srikanth, P. and K.N. Murthy, 2008. Named entity recognition for telugu. Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, (EAL' 08) pp: 41-50.

Sulaiman, S., R.A. Wahid, S. Sarkawi and N. Omar, 2017. Using stanford NER and illinois NER to detect malay named entity recognition. Int. J. Comput. Theory Eng.

Tjong Kim Sang, E.F. and F. De Meulder, 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL, (LLH' 03), pp: 142-147. DOI: 10.3115/1119176.1119195

Vijayakrishna, R. and S.L. Devi, 2008. Domain focused named entity recognizer for tamil using conditional random fields. Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian, Hyderabad, India, pp: 59-66.

Wang, Y. and J. Patrick, 2009. Cascading classifiers for named entity recognition in clinical notes. Proceedings of the Workshop on Biomedical Information Extraction, Sept. 18-18, Borovets, pp: 42-49.

Wei, L., 2004. Rapid development of hindi named entity recognition using conditional random fields and feature induction. ACM Trans. Computational Logic.