Named Entity Recognition using Support Vector Machine: A Survey

[1]DHANUSHYA.P, DHIVYA.S, KRITHIKA.H, SUGUNA RANI.R, [2]Mrs.ABINAYA

[1]UG scholar, NANDHA ENGINEERING COLLEGE

[2]ASSISTANT PROFESSOR, NANDHA ENGINEERING COLLEGE

Email Id:divyasharvagith@gmail.com

Abstract :
In this paper, a survey is done on various approaches used to recognize name entity in various Indian languages. Firstly, the introduction is given about the work done in the NER task. Then a survey is given about the work done in recognition of name entities in English and other foreign languages like Spanish, Chinese etc. In English language, lots of work has been done in this field, where capitalization is a major clue for making rules. Secondly, a survey is given regarding the work done in Indian Languages. There are various rule-based and machine learning approaches available for Named Entity Recognition. Named Entity Recognition (NER) is sub task of Information Extraction that includes identification of named entities and classification of them into named entity classes such as person, location and organization etc. NER can be used to preprocess textual information and convert it into structured form that can be useful for Information Retrieval, Machine Translation, Question Answering System and Text Summarization.

Index Terms - Named Entity, Named Entity Recognition, Tag set, Support Vector Machine (SVM)

I. INTRODUCTION

The term "Named Entity", the word Named refers the task to those entities for which one or many data stands as reference. It is widely used in Natural Language Processing (NLP). It is the subtask of Information Extraction (IE) where structured text is extracted from unstructured text. The task of Named Entity Recognition is to categorize all proper nouns in a document into predefined classes like person, organization, location, etc. NER has many applications in NLP like machine translation, question-answering systems, indexing for information retrieval, data classification and automatic summarization. It is two step process i.e. the identification of proper nouns and its classification. Identification is concerned with marking the presence of a word/phrase as NE in the given sentences and classification is for denoting role of the Support Vector Machines and Conditional Random Fields and Hybrid Approach. Although a lot of work has been done in English and other foreign languages like Spanish, Chinese etc with high accuracy but regarding research in Indian languages is at initial stage only. Here a survey of research done till now in English and other foreign and Indian languages are presented. Early systems are making use of handcrafted rule-based algorithms. While modern systems most often use machine learning techniques.

Whereas machine learning techniques uses a collection of annotated documents to train classifier for the given set of NE classes. According to the specification defined by MUC, the NER tasks generally work on seven types of named entities as listed below:

- Person Name
- Location Name
- Organization Name
- Abbreviation
- Time
- Term Name
- Measure

II. RELATED WORK

Basically NER originates from a set of earlier competitions organized within the Natural Language Processing (NLP) community. One of the most important is the Message Understanding Conference (MUC) where an earlier primary goal was to identify mentions or names of entities from unstructured news articles and classify them into predefined semantic categories.

On coarse level NER approaches are divided in into two branches: handcrafted rules and learning based methods. Methods based on handcrafted rules require developers to manually create extraction rules. Another approach, learning based uses machine learning techniques to accomplish named entity identification and its classification. NER is an enabling technology to

many applications. It is often used in a pre-processing step to many complex IE and IR tasks.

III. APPROACHES FOR NAMED ENTITY RECOGNITION

I . RULE BASED APPROACH

As mentioned earlier techniques for NER are most often divided into two main streams: handcrafted rules and learning based approaches. There are pros and cons of both the systems. Rule based techniques are very precise while learning based techniques give higher recall. Rule based techniques require small amount of training data as compare to learning based techniques where as learning based techniques need not require to build grammar. So based on application's requirements, appropriate technique can be chosen.

Methods based on handcrafted rules involve designing and implementing lexical-syntactic extraction patterns. They make use of existing information lists such as dictionaries that can frequently identify candidate named entities. An example of such rules can be „a street name is a multi-word phrase ends with the word „X' and proceeded by the preposition word „Y' ', where „X' and „Y' are lists of common words that are suitable for this purpose. For example, X could be Street' and Y could be in', thus the rule can recognize names of streets from texts such as „The Apple store in
Senapati Bapat Street in Pune'. Early entity recognition systems primarily adopted rule-based approaches.

II. LEARNING BASED APPROACHES

Machine learning is a way to automatically learn to recognize complex patterns or sequence labeling algorithms and make intelligent decisions based on data. Training examples or training data are usually an essential input to learning based methods. In machine learning, such annotated data are often called labeled data, which are often used to train an extraction model; on the other hand, the data without annotations are called test data.

Learning algorithms are methods able to consume features of training data to automatically induce patterns for recognizing similar information from unseen data. Learning algorithms can be generally classified into three types: supervised learning, semi-supervised learning and unsupervised learning. Supervised learning utilizes only the labeled data to generate a model. Semi-supervised learning aims to combine both the labeled data as well as useful evidence from the unlabeled data in learning. Unsupervised learning is designed to be able to learn without or with very few labeled data.

A. Supervised Methods

Supervised learning implies use of a program that can learn to classify a given set of labeled examples. These examples are made up of the same number of features. Different feature space is therefore used to represent each example. The learning process is called supervised, as labeled examples are used by the program to take right decision. Thus supervised learning approach requires preparing labeled training data to construct a statistical model, but it is unable to achieve a good performance without a large amount of training data

Supervised methods are class of algorithm that learn a model by looking at annotated training examples. Among the supervised learning algorithms for NER, considerable work has been done using Decision Trees, Hidden Markov Model (HMM), Maximum Entropy Models (MaxEnt), Support Vector Machines (SVM) and Conditional Random Fields(CRF). Typically, supervised methods either learn disambiguation rules based on discriminative features or try to learn the parameter of assumed distribution that maximizes the likelihood of training data.

Features are characteristic attributes of words designed for algorithmic purpose. Following features are most often used for the recognition and classification of named entities. These are defined into three categories i.e.

- Word-level features
- List lookup features
- Document and corpus features

IV. NAMED ENTITY RECOGNITION AND ITS APPROACHES

Name Entity Recognition:- Named Entity Recognition is the process of identification and classification of all proper nouns in a given text document or a sentence into predefined classes such as persons, locations, organizations, date, address and time expressions. Named Entities are defined as the proper names identified in a text. Identified text may be a person's names, organization's names, location's names, and date and time expressions. To make a computer acceptable and divide these named entities into pre-defined categories, which are important tasks of NLP. This task is defined as Named Entity Recognition. It is also called Information Extraction .

For example:-

Name entity type	**Examples**
ORGANIZATION	Global India
PERSON	President Pranab Mukherjee, Navneet
LOCATION	Chandigarh, Mount Everest
TIME	three fifty a m, 12:30 p.m.

Some of the supervised machine learning techniques is:

- Hidden Markov Model (HMM)
- Support Vector Machines (SVM)
- Conditional Random Fields (CRFs)

Hidden Markov Model(HMM):

HMM is the earliest model applied for solving NER problem. HMMs are generative models that proved to be very successful in a variety of sequence labeling tasks as Speech recognition, POS tagging, chunking, NER, etc. Inspite of these shortcomings the HMM approach offers a number of advantages such as a simplicity, a quick learning and also a global maximization of the joint probability over the whole observation and label sequences. There are seven types of named entities described in MUC. Fig shows examples of these named entities. By definition of the task, only a single label can be assigned to a word in context.

Type of Named Entity	Examples
person	Doctor, engineer, soldier, coach, etc
organization	Airline, company, college, news_agency, sports_team, etc
location	City, country, mountain, park, etc
product	Camera, engine, car, ship, etc
art	Written work, film, play
event	Election, natural_disaster, protest, sports_events, etc
building	Airport, dam, hospitals, library, etc

Fig 1. Tagset of named Entities

Support Vector Machine (SVM)

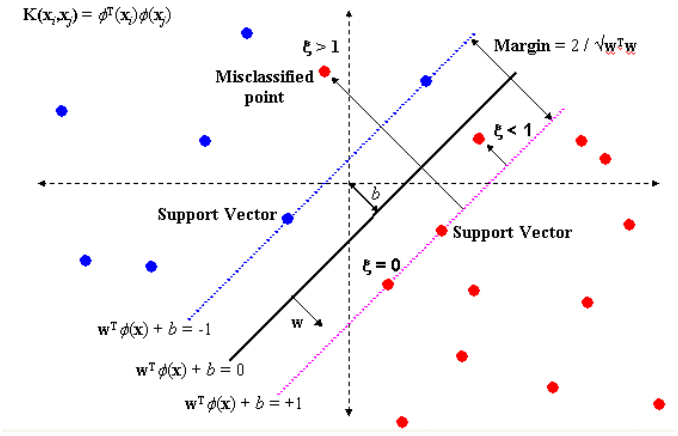
In a classification task using SVM the task usually involves training and testing data which consist of some data instances. The goal is to predict the class of the data instances. It is one of the famous binary classifier giving best results for fewer data sets and can be applied to multi-class problems by extending the algorithm. The SVM classifier used in the training set for making the classifier model and classify the testing data based on this model with the use of features.

SVM Based Models

Support Vector Machine was _rst introduced by Cortes and Vapnik (1995) based on the idea of learning a linear hyperplane that separate the positive examples from negative example by large margin. Large margin suggests that the distance between the hyperplane and the point from either instances is maximum. The points closest to hyperplane on either side are known as support vectors.

Figure-3 shows the geometric interpretation. The linear classifier is based on two parameters, a weight vector W perpendicular to the hyperplane that separates the instances and a bias b which determines the o_set of the hyperplane from the origin. A sample x is classified as positive instance if f(x) = wx+b > 0 and negative otherwise. If the data points are not linearly separable, then a slack is used to accept some error in classification. This prevents the classifier to over the data. When there are more than two classes, a group of classifiers are used to classify the instance.

McNamee and May_eld (2002) tackle the problem as binary decision problem,i.e. if the word belongs to one of the 8 classes, i.e. B- Beginning, I- Inside tag for person, organization, location and misc tags. Thus there are 8 classi_ers trained for this purpose. All feature used were binary. 258 orthography and punctuation features and 1000 language-related features were used. Window size was 7, that made the number of features used to 8806. To produce a single label for each token, the set S of possible tags were identified. If S was empty tag O was assigned else most frequent tag was assigned. If both beginning and inside tags were present then beginning tag was chosen. For CoNLL 2002 data, reported accuracies were 60.97 and 59.52 for Spanish and Dutch respectively.



Conditional Random Field (CRF):

It is a type of discriminative probabilistic model. It has all the advantage of MEMMs without the label bias problem. CRFs are undirected graphical models (also know as random field) which is used to calculate the conditional probability of values on assigned output nodes given the values assigned to other assigned input nodes

NER for Indian languages

NLP research around the world has taken major turn in the last decade with the advent of effective machine learning algorithms and the creation of large annotated corpora for various languages. But not much work has been done in NER for Indian languages because annotated corpora and other lexical resources have started appearing very recently in India. As common feature function like capitalization are not available in Indian languages and due to lack of large labeled dataset and lack standardization and spelling variation, so English NER cannot be directly used for Indian languages. So there arises the need to develop novel and accurate NER system for different Indian languages.

Characteristic and some problems faced by Hindi and other Indian languages

- No capitalization
- Brahmi script- It has high phonetic characteristic which could be utilized by NER system.
- Non-availability of large gazetteer
- Lack of standardization and spelling
- Number of frequently used words (common nouns) which can also be used as names are very large. "Also the frequency with which they can be used as common noun as against person name is more or less unpredictable."
- Lack of labeled data
- Scarcity of resources and tools
- Free word order language

In IJCNLP-08 workshop on NER for South and South East Asian languages, held in 2008 at IIT Hyderabad, was a major attempt in introducing NER for Indian languages that concentrated on five Indian languages- Hindi, Bengali, Oriya, Telugu and Urdu. The evaluation has reported F-Score of 44.91%. The development of a NER system for Bengali language is reported in 2008.

Its F-Score is 91.8%. The work of Gali et al, in 2008 reports lexical FScore of 40.63%, 50.06%, 39.04%, 40.94%, and 43.46% for Bengali, Hindi, Oriya, Telugu, and Urdu respectivel. In 2007 discussed the comparative study of Conditional Random Field and Support Vector Machine for recognizing named entities in Hindi language .

Indian languages are resource poor languages because of the non-availability of the annotated corpora, name dictionaries, good morphological analyzers etc. That is why high accuracy is not achievable yet.

The maximum accuracy for NER in Hindi is reported by Kumar and Bhattacharyya in 2006. They achieved an F measure of 79.7% using a Maximum Entropy Markov Model. Among other Indian languages, Punjabi language still lacks behind in this field. A research work is concentrated on NER for Punjabi language.

Punjabi is the official language of the Indian state of Punjab. It is also official language of Delhi and ranked 20th among the language spoken in the world. Among the Indian languages, Punjabi is the one in which the lots of research is going on in this field. Due to the nonavailability of annotated corpora, name dictionaries, good morphological analyzer etc. up to the required measure, Punjabi is the resource poor language like other Indian languages.

A recent research on NER for Punjabi language is done using Conditional Random Field (CRF) Approach. It was aimed to develop a standalone system based on CRF approach which can be used with other NLP applications like Machine Translation, Information Retrieval etc.

V. Conclusions

The Named Entity Recognition field has been thriving for more than fifteen years. It aims at extracting and classifying mentions of rigid designators, from text, such as proper names and temporal expressions. In this survey, we have shown the previous work done in English and other European languages. A survey is given on the work done in Indian Languages i.e. Telugu, Hindi, Bengali, Oriya and Urdu. An overview of the techniques employed to develop NER systems, documenting the recent trend away from hand-crafted rules towards machine learning approaches. Handcrafted systems provide good performance at a relatively high system engineering cost. When supervised learning is used, a prerequisite is the availability of a large collection of annotated data. Such collection are available from the evaluation forums but remain rather rare and limited in domain and language coverage. Recent studies in the field have explored semisupervised and unsupervised learning techniques that promise fast deployment for many entities types without the

prerequisite of an annotated corpus. Here also provided an overview of the evaluation methods that are in the use of NER accuracy. We have listed and categorized the features that are used in recognition of NE. The use of an expressive and varied set of features turns out to be just as important as the choice of machine learning algorithms.

VI. Future work

- The performance can further be improved by improving gazetteer lists.
- Analyzing the performance using other methods like Maximum Entropy and Support Vector Machines
- Comparing the results obtained by using different approaches and calculating the most accurate approach for it.
- Improve the performance of each NE tag to make it overall more accurate

References

[1] *Andrew Borthwick. 1999. "Maximum Entropy Approach to Named Entity Recognition" Ph.D. thesis, New York University.*

[2] Asif Ekbal, Sivaji Bandyopadhyay. "Bengali Named Entity Recognition using Support Vector Machine" in the proceedings of the IJCNLP-08 workshop on NER for South and South East Asian Languages, pages 51-58, Hyderabad, India.

[3] Asif Ekbal, Rejwanul Haque, Amitava Das, Venkateshwar Poka and Sivaji Bandyopadhyay. 2008., "Language Independent Named Entity Recognition in Indian Languages" in the proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 33-40, Hyderabad, India.

[4] Awaghad Ashish Krishnarao, Himanshu Gahlot, Amit Srinet, D.S.Kushwaha, 2009. "A Comparison of Performance of Sequential Learning Algorithms on the task of Named Entity Recognition for Indian Languages" in the proceedings of 9th International Conference on computer Science. Pages 123-132. Baton Rouge, LA, USA.

[5]. Nadeau, David, and Satoshi Sekine. "A survey of named entity recognition and classification." Linguistic Investigations 30.1 (2007): 3-26.

[6]. Srivastava, Shilpi, Mukund Sanglikar, and D. C. Kothari. "Named entity recognition system for Hindi language: a hybrid approach." International Journal of Computational Linguistics (IJCL) 2.1 (2011).

[7] Yuxiang Jia, Danqing Zhu, Shiwen Yu, "A Noisy Channel Model for Grapheme-based Machine Transliteration," Proceedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, pages 88–91, Suntec, Singapore, 7 August 2009. c 2009 ACL and AFNLP.

[8] Kamal Deep, Dr.Vishal Goyal, "Hybrid Approach for Punjabi to English Transliteration System," International Journal of Computer Applications (0975 – 8887) Volume 28– No.1, August 2011.

[9] Mitali Halder, Anant Dev Tyagi, "English-Hindi Transliteration by applying finite rules to data before training using Statistical Machine Translation," 978-1-4799-2845-3/13/$31.00 ©2013 IEEE.

[10] Deepti Bhalla, Nisheeth joshi, Iti mathur, "Improving the quality of machine translation output using novel name entity translation scheme," 987-1-4673-7/13/$31.00©2013 IEEE.

[11] Sujan Kumar Saha; Sanjay Chatterji; Sandipan Dandapat; Sudeshna Sarkar; Pabitra Mitra. "A Hybrid Approach for Named Entity Recognition in Indian Languages", In Proceedings of IJCNLP-08 workshop IIIT Hyderabad, India, January 2008, pp. 17-24.

[12] Sunita Sarawagi. 2008. "Information Extraction". "Indian Institute of Technology, CSE, Mumbai 400076, India.

[13]. Irmak, Utku, and Reiner Kraft. "A scalable machine-learning approach for semi-structured named entity recognition." Proceedings of the 19th international conference on World wide web. ACM, 2010.

[14]. Etzioni, Oren, et al. "Unsupervised named-entity extraction from the web: An experimental study." Artificial intelligence 165.1 (2005): 91-134

.

[15]. Isozaki, Hideki. "Japanese named entity recognition based on a simple rule generator and decision tree learning." Proceedings of the 39th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2001