JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN COMPUTER ENGINEERING

A HYBRID NAMED ENTITY RECOGNITION SYSTEM USING NATURAL LANGUAGE PROCESSING AND DECISION TREE LEARNING

[1]DASGUPTA SOUMI SUKHENDU, [2] PROF. AVANI R. VASANT

[1]M.E.[Computer Engineering] Student, Department Of Computer Engineering, V.V.P Engineering College, Rajkot, Gujarat
[2] Asst.Professor And Head, Department Of Information Technology, V.V.P Engineering College , Rajkot, Gujarat

dasguptasoumi@yahoo.co.in, avanivasant@yahoo.com

ABSTRACT: *Named Entity Recognition is classified as a subfield of Information Extraction and it is the core of Natural Language Processing (NLP) system. It helps machine to recognize proper nouns in text and associates it with appropriate types like person, location and organization, date and time, mentions of monetary amounts and percentages. This recognition of proper nouns and their extraction is significant in research areas like Information Retrieval, Machine Translation, Answering and Summarization Systems, Video Annotation, Semantic Web Search and Bioinformatics, Terrorism Event Extraction . Various approaches can be used for NER like Rule Based NER, Machine Learning Based NER and Hybrid NER. In this paper we propose a hybrid method using linguistic rules of natural language processing and supervised machine learning technique.*

Keywords— Named Entity(NE), Named Entity Recognition(NER), Message Understanding Conference(MUC),Natural Language Processing(NLP).

I: INTRODUCTION

Information Extraction is the process of analysing unstructured text and extracting the information relevant to some problem into a structured representation. The information relevant to a problem is usually divided into

1. Entities which can be persons, organizations or locations etc. that are located in the text.
2. Attributes that are related to the entities e.g. the title of the person or the type of the organization.
3. Facts i.e. the relations that exist between the entities.
4. Events in which entities participate

Named Entity Recognition (NER) actually deals with the task of retrieving names which can be categorized into classes which means structured text is extracted from unstructured text like newspaper. Named Entity Recognition can be classified as a sub problem of Information Extraction. The two stage process of Named entity Recognition consists of identification of proper nouns in the first stage and classification into the predefined categories of person , location , organization , date and time expressions or miscellaneous in the second stage . For example if New York is a named entity in the corpus it is necessary to identify the beginning and the end of this named entity in the sentence. Then the entity must be classified into the predefined category which is Named Entity Location in this case.

The term Named entity was first introduced in the Message Understanding Conference (MUC-6). MUC Conferences have an important contribution in the research area of Named Entity Recognition. In MUC-6 and MUC-7 Conferences the Named Entity task was recognized as categorizing the Named Entities into the following seven classes [1].

1. Person Name
2. Location Name
3. Organization Name
4. Abbreviation
5. Time
6. Term Name
7. Measure

The two broad approaches for developing a NER system are knowledge engineering approach and automatic training approach. Knowledge engineering approach is rule based and subject knowledge is required for named entity extraction. Automatic training approach is based on machine learning techniques and can be supervised, unsupervised or semi supervised. [3]

NER system can be domain specific which can be used for different business applications to extract named entities from free text. For example it can be an agro produce marketing domain for extracting entities like crop, name, variety, price, quantity, location, and deadline from the web. This can be done with the help of GATE framework which has a

information extraction system ANNIE comprising of different modules which help in the task of information extraction. NER system can also be used for research in biological domain. It can be used to automatically extract predefined entities like protein and DNA names. NER system can also be used to identify entities from juridical documents and named entities like location, organization, dates and document references can be identified.

II. RELATED WORK

Human beings can easily recognize named entities as proper names because of their capitalization but for machines it is difficult. A dictionary is not enough to classify the named entities into proper nouns because new proper nouns are being continuously added. Therefore it is difficult to update the dictionary with all these proper nouns continuously. The problem in NER is that they have an ambiguity related to the semantics of the sentence i.e. a proper noun can have different meaning according to the context. For example when is "April" a person name and when is it a month name? Another illustration is when is "White House" an organization name and when is it a location name?[2]

To solve these problems there are several classification methods which can be applied for NER tasks. Here we are using a supervised machine learning technique for Named Entity Recognition In supervised learning we have a training sample and our task to find a deterministic function which can map any input to a particular output and the possibility of any future disagreement should be minimized. Each element of the output space is called a class. Supervised Learning involves using a program which learns to classify a given set of labeled examples which should have same no of features. Supervised learning requires labeled training data to construct a statistical model. So a large amount of quality training data is needed for supervised learning. Supervised learning approaches are more expensive than unsupervised training methods in terms of time needed to preprocess the training data

The proposal in this paper is the use of decision tree induction as a solution to the problem of customizing a named entity recognition and classification (NERC) system to a specific domain. Decision Tree is a tree whose internal nodes are tests on input patterns and whose leaf nodes are categories of patterns. A decision tree assigns a class no or output to a input pattern by filtering the pattern down through the tests in the tree. Each test has mutually exclusive and exhaustive outcomes. Several systems for learning decision trees have been proposed and the prominent among those are ID3 and its new version, C4.5 and CART. Gyorgy Szarvas, Richard Farkas and Andras Kocsor proposed a system using Boosting and C4.5 decision tree algorithm which could achieve better F-measure on English and Hungarian test for the CoNLL corpus [8]. Georgios Paliouras1, Vangelis Karkaletsis1, Georgios Petasis1 and Constantine D. Spyropoulos proposed the use of a decision tree induction for the recognition of named entities person and organization and achieved good precision and recall.[6]

A NERC system assigns semantic tags to phrases that correspond to named entities e.g person, location and organization. Typically such a system makes use of two language resources: a recognition grammar and a lexicon of known names classified by the corresponding named entity types. NERC systems have been shown to achieve good results when the domain of the application is very specific. However the construction of the grammar and the lexicon for a new domain is a hard and time consuming process. Here the proposal is to use the decision trees as NERC grammars and the construction of these trees using machine learning. In order to validate our approach we tested C4.5 on the identification of person , location , organization and miscellaneous names. The results of the evaluation are very encouraging showing that the induced tree can outperform a grammar that was constructed manually.

III. PERFORMANCE EVALUATION

Definition and Scope

The primary motive in text mining is to detect names in the text that belong to task specific entity types. These tasks are called Named Entity Recognition because we try to recognize single or subsequent tokens in text which together makes a rigid designator phrase and determines the category to which these phrases belong. NER has been evaluated for various domains and languages. The named entities are classified into three main categories that is person, organization and location and its subtask consists of dividing it into entity names, temporal expressions and number expressions. There are three respective SGML tag elements which are ENAMEX, TIMEX and NUMEX. The Named Entities are listed below with their respective markup.

1. PERSON(ENAMEX)
2. ORGANIZATION(ENAMEX)
3. LOCATION(ENAMEX)
4. DATE(TIMEX)
5. TIME(TIMEX)
6. MONEY(NUMEX)
7. PERCENT(NUMEX)

Evaluation Metric

The NER task requires that the representation of the string should not lack thoroughness. Sometimes the right answer can be found using local pattern matching rules which is evident in most of the NUMEX expressions.While in other cases supposing we consider capitalization we cannot be sure whether it is a person or location.

Performance Evaluation metrics borrowed form the information retrieval community are

1. Precision (P): Precision is the fraction of the documents retrieved that are relevant to the user's information need.[1]

   $$P = \frac{\text{number of correct answers}}{\text{number of answers produced}} \quad (1)$$

2. Recall (R): Recall is the fraction of documents that are relevant to the query that are successfully retrieved.[1]

   $$R = \frac{\text{number of correct answers}}{\text{total possible correct answers}} \quad (2)$$

3. F-Measure: The weighted harmonic mean of precision and recall, the traditional F-measure or balance F-score is [1]

   $$\text{F-measure} = \frac{(\beta^2+1)PR}{(\beta^2 R + P)} \quad (3)$$

   β is the weighting between precision and recall typically $\beta=1$

4. F_1-measure : When precision and recall are evenly weighted that is $\beta=1$ F-measure is called F_1 measure [1]

   $$F_1\text{-measure} = \frac{2PR}{(P + R)} \quad (4)$$

Comparision

For comparison of various decision tree learning techniques the Precision, Recall and F-measure was calculated using the CONLL corpus and Adaboost decision tree learning algorithm [9]. Table-1 shows the results of these observed by Xavier Carreras, Lluis Marquez, Lluis Padro. Table-2 shows the results of C4.5 decision tree learning algorithm using data from the 6th message understanding conference observed by Georgios Paliouras,Vangelis Karkaletsis, Georgios Petasis and Constantine D. Spyropoulos .Table 3 shows the F-measure obtained by a hybrid decision tree learning method observed by Gyorgy Szarvas, Richard Farkas, Andras Kocsor.

Publication	Precision(%)	Recall(%)	F_β(%)
2003, Xavier Carraras etc. [5]	90.34	90.21	90.27
	83.19	85.07	84.12
	74.87	60.77	67.09
	74.89	63.16	68.45

Table.1. Results of Precision, Recall and F-measure for Adaboost decision tree learning algorithm

Publication	N.E	Recall (%)	Precision(%)
2000, Georgios Paliouras etc. [6]	Person	93.0	95.6
	Org	89.6	86.6

Table.2. Results of Precision, Recall for C4.5 decision tree learning algorithm

Publication	N.E	ENGLISH F-measure(%)	HUNGARIAN F-measure(%)
2006, Gyorgy Szarvas etc. [8]	LOC	93.43	95.07
	MISC	82.29	85.96
	ORG	88.32	95.84
	PER	96.27	94.67
	OVER-ALL	91.41	94.77

Table.3. Results of F-measure for hybrid decision tree learning algorithm

Discussion

In table 1 a Named Entity Extraction System for the CONLL Corpus the reviewed precision recall and F-measure is calculated for the named entities using AdaBoost decision tree learning algorithm. . In table 2 the reviewed precision and recall is calculated using the data from the 6th Message Understanding Conference and the system makes use of two language resources: a recognition grammar and a lexicon of known names classified by the corresponding named entity types. C4.5 was tested here on the identification of person and organization names. In table 3 the reviewed hybrid approach classifies NEs in the Hungarian and English languages by applying AdaBoostM1 and the C4.5 decision tree learning algorithm.. From the results we can predict that a hybrid approach can achieve good results for the person, location, organization names and miscellaneous entities.

IV. UNDERSTANDING C4.5 DECISION TREE LEARNING ALGORITHM

Given a tokenization of a test corpus and a set of n named entity categories the problem of named entity recognition can be reduced to the problem of assigning one of the 4n+1 tags to each token. For any particular N.E category x from the set of n categories the problem of named entity recognition can be reduced to the problem of assigning one of the 4n+1 tags to each token. For any particular N.E category x from the set of n categories, we could be in one of the 4 states: x start , x continue, x end, x unique. In addition a token could be tagged as other to indicate that it is not part of named entity. For instance we could tag the phrase [Jerry Lee Lewis flew to Paris] as [person start, person continue, person end, other, other, location unique].A decision tree can be considered to be composed of three elements.

1. Future: The possible outputs of the decision tree model
2. History: The information available to the model.
3. Questions: This is what a decision tree is all about. The objective of the decision tree algorithm is to to find the best sequence of questions to ask about the history to determine the future. In determining this sequence of questions the choice of the mth question to ask is determined by the answers to the previous m-1 questions.

C4.5 builds decision tree from a set of training data in the same way as ID3, using the concept of information entropy. The training data is a set $S=s_1,s_2$ of already classified samples. Each sample $si = x_1, x_2$ is a vector where x_1, x_2 represent attributes or features of the sample. The training data is augmented with a vector $C = c_1, c_2$ represent the class to which each sample belongs. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. Its criterion is the normalized information gain that results from choosing an attribute for spliiting the data. The attribute with the highest normalized information gain is chosen to make the decision.

The algorithm has few base cases.

- All the samples in the list belong to the same class. When this happens, itsimply creates a leaf node for the decision tree saying to choose that class.
- None of the features provide any information gain . In this case C4.5 creates a decision node higher up the tree using the expected value of the class.
- Instance of previously unseen class encountered. Again C4.5 creates a decision node higher up the tree using the expected value.

In pseudocode the general algorithm for building decision trees is

1. Check for base cases
2. For each attribute a
   Find the normalized information gainfrom spliiting on a
3. Let a_best be the attribute with the highest normalized information gain
4. Create a decision node that splits on a a_best
5. Recurse on the sublists obtained by splitting on a_best and add those nodes as children of node.

V. PROPOSED WORK

The system identifies and classifies NEs in the English language by applying C4.5 decision tree learning algorithm. The focus is on building as large a feature set as possible and use a split and recombine technique to fully exploit its potentials. This methodology provided an opportunity to train several independent decision tree classifiers based on different subsets of features and combine their decisions in a majority voting scheme.

To solve classification problems effectively it is worth applying various types of classification methods both separately and in combination. The success of hybrid methods lies in tackling the problem from several angles so algorithms of inherently different theoretical bases are good subjects for voting and for other combination schemes. Feature space construction and proper pre-processing of data also have a marked impact on system performance.

Steps for proposed model

1. Choose the relevant corpus or plain text.
2. First of tokenize the sentence and tag it.
3. The tagging is used to determine whether the token is a noun or a verb or belongs to any other part of speech.
4. A tree structure will be created for the sentence from which we can identify the proper nouns.
5. After extraction of the proper nouns we will go for feature extraction.
6. The decision tree algorithm now classifies the proper nouns into person, location and organization based on our feature extraction.
7. If we apply the same procedure on test data we can obtain similar results.
8. Majority voting can be applied for solving ambiguities and finally performance is judged on the basis of precision, recall and f-measure values.

VI. CONCLUSION AND FUTURE WORK

The text will be classified into four classes person, location, organization and miscellaneous. The f-measure of all the classes should be high to obtain good performance of the system. The future work of the system consists of generation of a parse tree which depicts the POS tagging associated with the structure of the sentence. The next aim will be to extract the proper nouns from the parse tree and generate a rich feature set which can be used for decision tree learning. The final step consists of classification of the proper nouns into person, location, organization and miscellaneous with the help of decision tree learning and judging the performance on the basis of Precision, Recall and F-measure.

REFRENCES

[1] Darvinder Kaur, Vishal Gupta, "A Survey Of Named Entity Recognition in English and other Languages", IJCSI, Vol 7, Issue-6, November 2010.

[2] Alireza Mansouri, Lilly Suriani Affendy, Ali Mamat, "Named Entity Recognition Using a New Fuzzy Support Vector Machine", IJCSNS, Vol.8 No-2, February 2008.

[3] Priyanka Joshi, Sanjay Chaudhury, Vikas Kumar, "Information Extraction from Social Network for Agro-produce Marketing, DAIICT, Gandhinagar, Gujarat, India

[4] Veronica S. Moertini, "Towards The use of C4.5 algorithm for classifying Banking Dataset",Integral Vol 8 No 2 October 2003 .

[5] Xavier Carreras, Lluis Marquez, Lluis Padro,"A Simple Named Entity Extractor Using AdaBoost",Proceedings of the seventh conference on Natural Language Learning at HLT-NAACL 2003 Volume 4,Stroudsburg,PA,USA,2003.

[6]Georgios Paliouras,Vangelis Karkaletsis, Georgios Petasis and Constantine D. Spyropoulos, "Learning Decision Trees for Named Entity Recognition and Classification, ECAI 2000 .

[7] Hideki Isozaki," Japanese Named Entity Recognition based on a Simple Rule Generator and Decision Tree Learning.", ACL 01Proceedings of the 39th Annual Meeting on Association Of Computational Linguistics, Stroudsburg, PA, USA, 2003.

[8] Gyorgy Szarvas, Richard Farkas, Andras Kocsor, "A Multilingual Named Entity Recognition System Using Boosting and C4.5 Decision Tree Learning Algorithm", Springer-Verlag, Berlin, Hiedelberg, 2006.

[9] Steven Bird, "NLTK: The Natural Language Toolkit", Proceedings Of The COLING/ACL 2006 Interactive Presentation Sessions, Sydney, July-2006

[10] Dan Klein, Christopher D. Manning, "Parsing with TreeBank Grammars: Empirical Bounds, Theoretical Models, and the structure of Penn TreeBank", Proceeding ACL '01 Proceedings of the 39th Annual Meeting on Association for Computational Linguistics,2001.