Domain Specific Named Entity Recognition

Ashwini A. Shende

Department of Computer Science
SRKNEC
Nagpur , India

zashwini@rediffmail.com

Avinash J. Agrawal

Department of Computer Science
SRKNEC ,
Nagpur , India

avinashjagrawal@gmail.com

Abstract — **Named entities are phrases denoting the names of persons, locations, organizations etc. in text documents. These phrases are important for the access to document content, since they form the building blocks for the analysis of documents. Named Entity Recognition has applications in Natural Language Processing, document indexing, document annotation, translation, etc. NER plays an important role in various research areas of Natural Language Processing (NLP) like Question Answering and Summarization Systems, Information Retrieval, Machine Translation, Video Annotation, Semantic Web Search, Bioinformatics etc. The computational research of automatically identifying named entities in texts forms a vast and heterogeneous pool of strategies, methods and representations. In this paper, we will present an overview of the various methods used for implementing NER systems by giving the merits and demerits of each. We will also discuss some of the approaches suggested and implemented by NER system developers.**

Keywords— **Named Entity; Training; Learning; Word Disambiguation; Context**

I. INTRODUCTION

The term "Named Entity", widely used in Natural Language Processing, was coined for the Sixth Message Understanding Conference (MUC-6). Named Entity (NE) can be a named: location, person, organization, date, time, etc. characterized by instances.

In the expression "Named Entity", the word "Named" aims to restrict the task to only those entities for which one or many rigid designators stands for the reference. For instance, *the automotive company created by Henry Ford in 1903* is referred to as *Ford* or *Ford Motor Company*. Rigid designators include proper names as well as certain natural terms like biological species and substances. There is a general agreement about the inclusion of temporal expressions and some numerical expressions such as amounts of money and other types of units. While some instances of these types are good examples of rigid designators (e.g., *the year 2001* is the 2001st year of the Gregorian calendar) there are also many invalid ones (e.g., *in June* refers to the month of an undefined year – *past June*, *this June*, *June 2020*, etc.).

A Named Entity (NE) is found in texts accompanied by contexts: i.e. words that are left or right of the NE. The NER task mainly aims at identifying contexts inducing the NE's nature. E.g. the occurrence of the word "*President*" in a text means that this word or context may be followed by the name of a president as President "*Obama*". Likewise, a word preceded by the string "*footballer*" induces that this is the name of a footballer. Named Entity recognition may be viewed as a classification problem where every word is assigned to a NE class according to the context. The perfect NER system should be capable of identifying and classifying the contexts that are most relevant to recognize a NE.

Implementing NER Manually is quite simple, as generally named entities are proper names and most of them have initial capital letters and can be easily recognized. But for machines, it is quite difficult. One simple technique is classifying named entities using Dictionaries .But over a period of times new proper nouns are getting created continuously. So it is practically impossible to add all proper nouns to a dictionary. Deciding upon the sense of the named entity is another critical issue. Most of the problems in NER are because of semantic (sense) ambiguity, Also proper noun has different senses according to the context. For illustration, when is "*The White house*" an organization, and when is it a location? When is "*June*" a person name? And when is it a month name? Or in "He visited Bush at *White House*", here *White House* is a location", but in "*White House* announced the list of ministry candidate", *White House* is an organization.

NER is considered to be a subproblem of Information Extraction. It needs to process structured as well as unstructured text documents and identify expressions that refer to named entities .NER is the core of natural language processing techniques. NER task works in two phases, first phase does Identification of proper names in text and second phase does the Classification of these names into a set of domain specific, predefined categories.

The NER system is implemented in three steps. The first step is training the corpus collection. This step builds an initial corpus containing text documents. This corpus is called **learning corpus**.

The second step is context extraction and classification,. Here the goal is to reveal contextual NE in a document corpus. A context considers words surrounding the NE in the sentence

in which it appears. It is a sequence of words, that are left or right of the NE. In many cases, the same context can introduce different NE .The goal is to find high-quality context.

Third step builds NE recognition model. The flow diagram of NER system is shown below.

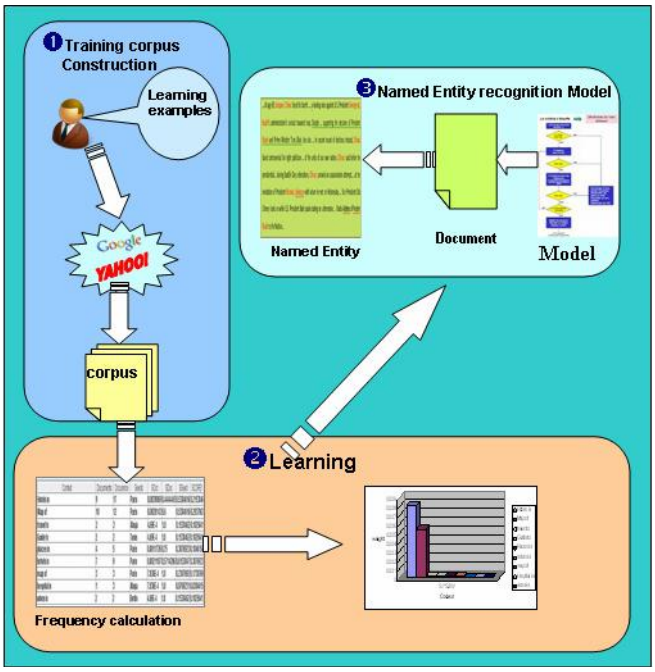

FIG 1

NER FLOW DIAGRAM

NER systems largely rely on **Lexical resources** and **Syntactical patterns** for classification. **Language-Specific** & **Domain-Specific tools can be** used for this purpose. Language-Specific Tools are Part-of-speech tags, Noun phrase tags, syntactic tags ,Grammar rules, Affix information (character n-grams) ,Orthographic patterns, Lexical features ,Punctuation & parentheses handling ,Word triggers, word roots, word variations etc. Domain-Specific Tools are Specialized dictionaries**,** Gazetteers (reference information), Bag of words , Definition of rules describing entities and their possible contexts ,Cascaded entities and Other external resources etc.

For NER task**,** large collections of documents (called **training corpus**) are analysed to obtain sufficient knowledge for designing rules or for feeding machine learning algorithms. NER System deals with two types of objects, **Concepts** and **Instances**. A **Concept** represents a set of thing of domain interest that have something in common while an i**nstance** is a single example of a concept .e.g. *human* is a concept while *Shakespeare* is an instance of that concept. Knowledge base developed from training corpus is used for learning process.

NER systems are normally domain specific. Performance degrades when test domain differs from training domain. Considerable effort is required to perform well in a new domain .The basic aim of NER is to extract and classify names into some particular categories from given corpus with respect to the context of names. Incorporating language or domain-specific knowledge requires additional pre and or post processing.

A NER Task evaluation uses the information-retrieval terms like **Precision** (P) and **Recall (R)** Precision and recall are combined to form one measure of NER performance i.e. the **F-measure,** which is computed by the uniformly weighted harmonic mean of precision and recall.

Researchers use different methods such as **Rule-based** NER, **Machine Learning-based** NER and **Hybrid** NER to identify Named Entities from text. Hand-made Rule-based NER consist of a set of patterns using grammatical, syntactic and orthographic features in combination with dictionaries for named entity recognition. This approach rely on manually coded rules and manually compiled corpora and produce better results for restricted domains but lacks portability and robustness. This approach is often domain and language specific and do not adapt well to new domains and languages.

In Machine Learning-based methods, system look for patterns and relationships in text to make a model using statistical techniques and machine learning algorithms. Three types of machine learning methods namely **Supervised**, **Semi-supervised** and **Unsupervised**. A **Supervised Learning** method perform tagging of words of a test corpus when they are annotated as entities in the training corpus. Supervised learning require large amount of training data for good performance. The main technique for **Semi-supervised learning** is **bootstrapping** and involves a small degree of supervision such as a set of seeds for starting the learning process. **Unsupervised** learning is without any feedback. It uses the clustering approach. Named entities can be gathered from clustered groups based on the similarity of context. The **Unsupervised learning** techniques rely on lexical resources, lexical patterns and statistics computed on a large unannotated corpus. This approach can be easily ported to different domain or languages.

In Hybrid NER system, the approach is to combine rule based and machine learning-based methods and make new methods by getting the benefits of both. Although this type of approach can get better result than other approaches, but the weakness of handcrafted Rule-base NER remains there.

II. RELATED WORK

M. Collins and Singer (1999) suggested a Rule based algorithm for named entity classification, based on the word meaning disambiguation and exploits the redundancy in the contextual characteristics. This system operates on a large corpus to produce a generic list of proper nouns. The names are collected by searching for a syntax diagram with specific properties. E.g. a proper name is a sequence of consecutive words, within a noun phrase, that are tagged as NNP or NNPS

by a part-of-speech tagger and in which the last word is identified as the head of the noun phrase.

M. Collins and Singer parse a complete corpus in search of candidate NE patterns. A pattern is a proper name followed by a noun phrase in apposition (e.g., *Maury Cooper, a vice president at S&P*). Patterns are kept in pairs {*spelling, context*} where *spelling* refers to the proper name and *context* refers to the noun phrase in its context. Starting with an initial seed of spelling rules (e.g., *rule 1*: *if the spelling is "New York" then it is a Location; rule 2: if the spelling contains "Mr." then it is a Person; rule 3: if the spelling is all capitalized then it is an organization*), the candidates are examined. Candidate that satisfy a spelling rule are classified accordingly and their contexts are accumulated. The most frequent contexts found are turned into a set of contextual rules. The steps above contextual rules can be used to find further spelling rules, and so on.

E. Alfonseca and Manandhar (2002) studied the problem of labeling an input word with an appropriate NE type. NE types are taken from WordNet (e.g., location>country, animate>person, animate>animal, etc.). Their approach is to assign a topic signature to each WordNet synset by merely listing words that frequently co-occur with it in a large corpus. Then, given an input word in a given document, the word context (words appearing in a fixed-size window around the input word) is compared to type signatures and classified under the most similar one.

E. Alfonseca and Manandhar proposed a method for Resolving named entity ambiguity using unsupervised approach. Generally unsupervised approach is formulated as a clustering problem. In clustering, target is to group mentions of the same entity into the same cluster. A different approach is suggested based on the **Distributional Hypothesis** and **edit distance** which associates an ambiguous entity to its corresponding entry in the knowledge base or gazetteer list. Suggested approach experimented with two types of contextual features ,**bag-of-words** and **bigrams** as well as the **edit distance**. It is proved that the combination of these types of knowledge offered a superior performance than each one individually or any subset of them, leading to the conclusion that they are able to capture non-overlapping information essential for resolving named entity ambiguity.

The *KNOWITALL* system planned by **Etzioni (2005)** aims at automating the process of extracting named entities from the Web in an unsupervised and scalable manner. This system is not intended for recognizing a named entity, but used to create long lists of named entities. It is not designed to resolve the ambiguity in documents.

Etzioni also suggested distinct ways to improve the system performance. First suggested method was **Pattern Learning** that learns domain-specific extraction rules enabling additional extractions. Second method was **Subclass Extraction** which automatically identifies sub-classes in order to boost recall. Third suggested method was **List Extraction** which locates lists of class instances, learns a "*wrapper*" for each list, and extracts elements of each list. Since each of the above mentioned methods bootstraps from *KNOWITALL*'s domain independent methods, the methods also obviate hand-labelled training examples. These methods gave *KNOWITALL* superior performance.

David Nadeau, Peter D. Turney and **Stan Matwin (2006)** suggested a system for recognizing named entities. Their work is based on the work done by **Collins** and **Etzioni** .The system exploits human-generated HTML markup in Web pages to generate gazetteers, then it uses simple heuristics for the entity disambiguation in the context of a given document.

David Nadeau, Peter D. Turney and **Stan Matw** proposed a named-entity recognition system that combines named entity extraction with a simple form of named-entity disambiguation. Technique used for the system is general enough to be applied to other named-entity types and advances the state-of-the-art of NER by avoiding the need for supervision and by handling novel named-entity types.

The proposed system architecture is made of two modules. The first module is used to create large gazetteers of entities. The second module uses simple heuristics to identify and classify entities in the context of a given document (i.e., *entity disambiguation*).

Alireza Mansouri, Lilly Suriani Affendey, Ali Mamat (2008) presented a review of different approaches used for Named Entity Recognition. All the methods and models mentioned ,tried to improve precision in recognition module and portability in recognition domain, as one of the important problem of NER is to change and switch data domain to new domain called **portability.**

In the Rule-based method, there was improvement in precision by adding more rules and developing grammatical rules, however portability was reduce automatically, because of fix rules and method constructors.

TABLE I

RESULTS OF EXPERIMENT WITH HAND-MADE RULE NER SYSTEM

	System	R	P	F(β=1)
1	IsoQuest,Inc	90	93	91.60
2	NYU System	86	90	88.19
3	U. of Manitoba	85	87	86.37

TABLE II

RESULTS OF EXPERIMENT WITH MACHINE LEARNING BASED NER SYSTEM

	System	R	P	F(β=1)
1	MENE	89	96	92.20
2	IdentiFinder	89	92	90.44
3	Association Rule Mining	66.34	83.43	70.16

TABLE III

RESULTS OF EXPERIMENT WITH HYBRID NER SYSTEM

	System	R	P	F(β=1)
1	LTG	92	95	93.39
2	NYU Hybrid	85	93	88.80

III. CONCLUSION AND FUTURE WORK

In this paper we have given an overview of the techniques employed to develop NER systems, concluding that the recent trend moves 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 collections are available from the evaluation forums but remain rare and limited in domain and language coverage. Recent studies in the field have explored semi-supervised and unsupervised learning techniques that promise fast deployment for many entity types without the prerequisite of an annotated corpus.

NER system needs to consider certain issues like Semantic (sense) ambiguity, Context ambiguity,common word ambiguity as well as issues of style, structure, domain etc .Generally nouns in text document represent Named entities. Common nouns represent Concepts and proper nouns represent Instances. Distinguishing between concepts and instances is very important for named entity recognition as concepts and instances can be used in different ways in a language. However some entities can act as both concept and instance.

The efficiency of NER system can be measured from the accurate classification of named entities. Entity classification needs to address ambiguity issues. Syntactical and Contextual features can help to some extend for resolving ambiguity. In some cases Statistical methods can also be used. Combination of all such features can give better performance.

Generation of large gazetteer lists from the training corpus is another evaluation parameter of NER systems. Larger the list size more accurate is the entity classification. Additional techniques can be used along with extraction methods to generate large lists.

We will be focusing on the use of machine learning approach for NE recognition. Our aim is to uncover NE in a document corpus, accompanied by contexts: Contexts that occur with given learning examples can be extracted from text document corpus. Different weighting measures can be used to classify the contexts in order to identify the most pertinent contexts for the recognition of a NE. This classification enables to derive a model for NE recognition. Same technique can be applied to multiple entity types. One of the future work that we will recommend is to measure similarity between contexts. This can be used to cluster similar contexts.

REFERENCES

[1] Alireza Mansouri, Lilly Suriani Affendey, Ali Mamat, "Named Entity Recognition Approaches" IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.2, February 2008

[2] Enrique Alfonseca and Suresh Manandhar, "An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery" In Proceedings of the 1 st International Conference on General WordNet , 2002

[3] Enrique Alfonseca and Suresh Manandhar, "Distinguishing concepts and instances" In Proceedings of the International Conference, 2002

[4] David Nadeau, Peter D. Turney and Stan Matwin "Unsupervised Named-Entity Recognition: Generating Gazetteers and Resolving Ambiguity", In Proceedings of the 19th Canadian Conference on Artificial Intelligence, 2006

[5] Collins, Michael and Y. Singer. "Unsupervised models for named entity classification", In proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 1999.

[6] J.Kim, I. Kang, k. Choi, "Unsupervised Named Entity Classification Models and their Ensembles", Proceedings of the 19th international conference on Computational linguistics, 2002

[7] Ioannis P. Klapaftis ,Suresh Manandhar, "Unsupervised Named Entity Resolution "Proceedings of the 3rd IEEE International Conference on Multimedia Communications, Services and Security, 2010

[8] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria PopescuTal Shaked, Stephen Soderland, Daniel S.Weld, and Alexander Yates "Unsupervised Named-Entity Extraction from the Web: An **Experimental** Study", Published in Journal Artificial Intelligence , Volume 165 ,Issue 1 ,2005

[9] WAHIBA BEN ABDESSALEM KARAA "NAMED ENTI RECOGNITION USING WEB DOCUMENT CORPUS" International Journal of Managing Information Technology (IJMIT) Vol.3, No.1, February 2011

[10] R. Sirhari, C. Niu, W. Li, "A Hybrid Approach for Named Entity and Sub-Type Tagging" Proceedings of the sixth conference on Applied natural language processing ,Acm Pp.247 - 254 , 2000.