ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

Approaches to Named Entity Recognition: A Survey

Ms. Maithilee L. Patawar, Mrs. M. A. Potey

M. E. Student, Dept. of Computer, D. Y. Patil College of Engineering, Akurdi, Pune, India

H. O. D., Dept. of Computer, D. Y. Patil College of Engineering, Akurdi, Pune, India

ABSTRACT: This paper surveys on different approaches used for Named Entity Recognition (NER) which identifies named entities from data. Basic terms related to NER and Named Entity Recognition and Classification (NERC) is introduced. Various approaches from previous work including rule based techniques and learning based methods are thoroughly reviewed and explained. Traditionally rules were more common in named entity identification. Introduction of Named Entity Rule Language made the task easier. Learning based approaches are then widely used due to better results. Role of these approaches in various applications has reviewed and their suitability in particular scenario is discussed. In addition to these approaches, recent hybrid approaches for NER which attempt to integrate former approaches are also explained. Such hybrid approaches are particularly designed for specific application or language.

KEYWORDS: Named Entity, Named Entity Recognition, Named Entity Rule Language, Tagset, Gazetteers.

I. INTRODUCTION

The term Named Entity Recognition can be considered as subtask of Information Extraction where structured text is extracted from unstructured text. It seeks to locate and classify elements in text into predefined categories. In the term "Named Entity", the word Named restricts the task to those entities for which one or many rigid designators stands as referent. This is widely used in Natural Language Processing (NLP).

The task of Named Entity Recognition was formally defined in Message Understanding Conference 6 (MUC6) as the task of identifying the names of all the people, organizations and geographic locations in a text, as well as time, currency and percentage expressions. Since MUC6 there has been increasing interest in this topic and extensive effort has been devoted into its research. Major computational linguistic conferences hosted special tracks for the task and there has been steady growth of publications throughout the years. Several events made the attempt to enrich the definition of the task. For example, MUC7 included date and time entities, and introduced the multilingual named entity recognition. The Automatic Content Extraction (ACE) program introduced several new entity types and a more fine-grained structure of entity subtypes in an attempt to achieve more precise classification of entities, such as distinguishing government, educational and commercial organizations from each other, which all belong to the coarse-grained entity type 'organization'. The task has also been extended to technical domains to recognize domain-specific entities, typically in the domain of biomedical science to recognize domain-specific entities such as gene and protein names. Large amount of resources have been created for the purpose of evaluating biomedical entity recognition and successive events have been hosted to motivate the research.

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. In brief, an entity is a unique real word object, a mention or name is a lexicalized expression used to designate an entity, and a semantic category is a high level concept that groups same types of entities, such as 'people' , 'place' and 'organization'. This task is called Named Entity Recognition (NER), a term first coined at the sixth Message Understanding Conference (MUC6). In addition to IE, NER is also an important technology for many other applications and research areas.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

The NER task naturally translates into two subtasks: name detection or identification that finds the boundaries of entity names; and semantic classification that assigns the most appropriate semantic category. Multiple approaches are proposed for NER which simplifies these tasks.

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. These rules are usually expressed as lexico-syntactic patterns and semantic constraints that determines the occurrences of similar named entities. Another approach, learning based uses machine learning techniques to accomplish named entity identification and its classification. These two approaches are reviewed in next sections.

NER is an enabling technology to many applications. It is often used in a pre-processing step to many complex IE and IR tasks. Tasks like relation extraction, knowledge base generation, question answering, etc needs to identify named entities first to complete the tasks [5]. So in such cases NER is the first step of processing. NER is also used in improving semantic search.

III. 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. They are efficient for domains where there is certain formalism in the construction of terminology[17]. A typical example is the biology domain, where certain types of entities can be extracted by domain-specific rules with sufficient accuracy. Also it has been successfully applied in open information extraction where information redundancy is available for relatively simple types of entities.

However, the major limitation of these systems is that they require significant expertise from the human developers, in terms of the knowledge about the language, domain as well as programming skills. These knowledge and resources are often expensive to build and maintain and are not transferable across domains [17]. Consequently these approaches suffer from limited or no portability.

A simple rule based example can be discussed here based on rule based: In general regular expressions provide a flexible way to match strings of text, such as particular characters, words, or patterns of characters. Suppose an NER system is looking for a word that

1. Starts with a capital letter “A”
2. Is the first word on a line
3. The second letter is a lower case letter
4. Length is exactly of 3 letters

then the regular expression would be “$A[a-z][aeiou]" where,
[a-z] - any letter in range a to z
[aeiou] - any vowel
$-indicates the beginning of the string

This is simple example on rule based approach. Traditionally, rule-based NER systems were based on the popular CPSL cascading grammar specification. CPLS is designed in such way that rules following standards can be efficiently used and executed. As per the standard, an execution model is based on strict left-to-right execution. At most one rule is used to match text area at a time provided annotations should not overlapped. But rigidity of the rule matching semantics makes it difficult to express operations frequently used in rule-based information extraction. To overcome these limitations several declarative algebraic languages have been proposed for rule-based IE systems. These languages are not constrained by the requirement that all rules map onto finite state transducers, and therefore can

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

express a significantly richer semantics than grammar based languages.

Many languages like AQL, CPSL and JAPE were developed which are useful for general purpose information extraction. These are suitable for general rule based information extraction but not adequate to implement NER rules. To make NER rules easier to develop and to understand, NERL (Named Entity Rule Language)[17] was developed. NERL is a declarative rule language designed specifically for named entity recognition. A NERL rule creates an intermediate concept or named entity (IntConcept for short) by applying a NERL rule on the input text and zero or more previously defined intermediate concepts. It is observed that NERL provides a higher level abstraction catered specifically towards NER tasks, thus hiding the complexity inherent in a general-purpose IE rule language. In doing so, NERL restricts the large space of operations possible within a general-purpose language to the small number of predefined "templates". Generally rule based approaches are prone to both overreach and skipping over named entities. So they used as classifiers in machine-learning approaches, or as candidate taggers in gazetteers.

IV. 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. Central to the machine learning paradigm is the idea of providing positive and negative training examples for the task; modeling distinctive features associated with examples; and design algorithms that consume these features to automatically distinguish positive from negative examples and to recognize similar information from unseen data. Training examples or training data are usually an essential input to learning based methods. They often take the form of annotations that are labeled instances of named entities, created by domain experts in a document annotation process. 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. In many unsupervised learning methods that do not require annotations, a set of 'seed data' is often needed to support the learning. Seed data are typically lists of example entities of a particular type. Essentially they can be considered as training data in a rather different form. Features are characteristics of text objects to be studied in a computational linguistic problem. In NER, the target text objects are tokens (e.g., words) or sequences of tokens for identification and classification[3]. Features are used to create a multidimensional representation of the text objects, which can then be used by learning algorithms for generalization in order to derive patterns that can extract similar data and distinguish positive from negative examples.

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.

Methods under these are more elaborated in following sections:

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. This is due to data sparseness problem arise if small amount of training data is used. In recent years several statistical methods based on supervised learning method were proposed.

As discussed 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[15], Hidden Markov Model (HMM)[19], Maximum Entropy Models (MaxEnt)[7], Support Vector Machines (SVM)[18] and Conditional Random Fields(CRF)[2]. 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.

a. Hidden Markov Model

HMM is the earliest model applied for solving NER problem. HMMs are generative models that proved to be very

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

successful in a variety of sequence labeling tasks as Speech recognition, POS tagging, chunking, NER, etc. Its purpose is to maximize the joint probability of paired observation and label sequences. If, besides a word, its context or another features are taken into account the problem might become intractable[6]. Therefore, traditional HMMs assume an independence of each word from its context that is, evidently, a rather strict supposition and it is contrary to the fact. In spite 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. IdentiFinder was an earliest HMM model based developed system to detect NER. There are seven types of named entities described in MUC. Fig 1 shows examples of these named entities[19]. 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

In HMM, for each of the regions, a model for computing the likelihood of words occurring within that region (name-class) was used called as statistical bigram language model. A statistical bigram language model computes the likelihood of a sequence of words by employing a Markov chain, where every word's likelihood is based simply on the previous word. More formally, every word is represented by a state in the bigram model, and there is a probability associated with every transition from the current word to the next word. The use of a statistical bigram model in each name-class means that the number of states in each of the name-class region is equal to vocabulary size, |v|.

For the purposes of name-finding, it is needed to find the most likely sequence of nameclasses (NC) given a sequence of words (W):

$$\max \Pr(NC|W) \tag{1}$$

Here an assumption of generative model i.e. that the HMM generates the sequence of words and labels. By Bayes' rule:

$$\Pr(NC \mid W) = \frac{\Pr(W, NC)}{\Pr(W)} \tag{2}$$

Since the unconditioned probability of the word sequence, the denominator is constant for any given sentence, the right hand side of equation 4 can be maximized by maximizing numerator alone. The numerator of Equation 4 is the joint probability of the word and name-class sequence. As is necessary with a Markov model, independence assumptions are made while computing this joint probability[6]. Accordingly, the generation of words and name-classes proceeds in three steps:

1. Select a name-class NC, conditioning on the previous name-class and the previous word
2. Generate the first word inside that name-class, conditioning on the current and previous name-classes.
3. Generate all subsequent words inside the current name-class, where each subsequent word is conditioned on its immediate predecessor (as per a standard bigram language model).

These three steps are repeated until the entire observed word sequence is generated.

b. Maximum Entropy based model

Maximum entropy is a very flexible method of statistical modeling which turns on the notion of “\futures”,

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

“\histories”, and “\features”. Futures are defined as the possible outputs of the model. A maximum entropy solution to this, or any other similar problem allows the computation of p(f|h) for any f from the space of possible futures, F, for every h from the space of possible histories, H. As with decision trees, a “history" in maximum entropy is all of the conditioning data which enables to assign probabilities to the space of futures[7]. In the named entity problem, reformulation can be done in terms of finding the probability of f associated with the token at index t in the test corpus as:

$$p(f|h_t) = p(f|\text{Info. derivable from test corpus wrt token t}) \tag{3}$$

Given a set of features and some training data, the maximum entropy estimation process produces a model in which every feature g_i has associated with it a parameter α_i[7]. This allows to compute the conditional probability as follows:

$$p(f \mid h) = \frac{\prod_i \alpha_i^{g_i(h,f)}}{Z_\alpha(h)} \tag{4}$$

$$Z_\alpha(h) = \sum_f \prod_i \alpha_i^{g_i(h,f)} \tag{5}$$

This equation tells that the conditional probability of the future given the history is the product of the weightings for all features which are active on the <h, f> pair, normalized over the products for all the futures.

Maximizing the entropy ensures that for every feature g_i, the expected value of g_i, according to M.E. model will be equal to empirical expectation of g_i in the training corpus. Finally, Viterbi algorithm is used to find the highest probability path through the trellis of conditional probabilities which produces the required valid tag sequences.

c. CRF based model

As Conditional Random Fields (CRFs) are a class of statistical modeling method often applied in pattern recognition and machine learning, they are used for structured prediction. Whereas an ordinary classifier predicts a label for a single sample without regard to “neighboring" samples, a CRF can take context into account. The model uses sequence modeling algorithms which are probabilistic in nature. Sequence labeling is a type of pattern recognition task that involves the algorithmic assignment of a categorical label to each member of a sequence of observed values. A common example of a sequence labeling task is part of speech tagging, which seeks to assign a part of speech to each word in an input sentence or document. Sequence labeling can be treated as a set of independent classification tasks, one per member of the sequence[21].

Because of the strong ability to integrate any kind of features which plays an important role during training, CRFs becomes one of the key factors affecting the NER performance. The features of CNER include not only the internal features from context, such as character information, POS and boundary, but also the external features based on the statistical results such as surname that the prefix of family names, the suffix of location and organization and so on. In addition, the feature template is also found to play an important role in CRF based NER.

To define the characteristic function, the true character of observations on the b(x, i) collection is firstly constructed. The characteristic collection not only demonstrates priori distribution of the training data, but also reflects the model distribution. As to the specified values from the current state which equal to the state function or from the state between the previous and the current which corresponding the transfer function, the value of each characteristic can be assigned as an observation characteristic function b(x, i) shown as follows:

$$f(y_{i-1}; y_i; x; i) = b(x,i) \tag{6}$$

else

$$f(y_{i-1}; y_i; x; i) = 0 \tag{7}$$

where b(x,i) represents the value of a real observation. According to different attributes and requirements from the number of label sets of training file format, feature function template sets can be chosen.

As these types on NER systems need training first, training feature set of CRFs is a very important parameter, which can directly affect the NER results. The main features of CRFs can be divided into two categories: the internal and the external. The former includes character or word, POS, boundary and other context information. Although these basic internal features may have certain effects on the results, the external features are needed to obtain better results. The external features are mainly from the general statistical information of corpus, which consists of the prefix of common family names, the suffix of place names and organizational names and so on.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

Supervised methods give best performance but needs large annotated data for training purpose. This annotation need to be done manually by expert and is very time consuming. To overcome these problems, semi-supervised and unsupervised techniques are used for NER task.

B. Semi-supervised Methods

Semi supervised learning algorithms use both labeled and unlabeled corpus to create their own hypothesis. Algorithms typically start with small amount of seed data set and create more hypothesis' using large amount of unlabeled corpus. It make use of unlabeled data for training - typically a small amount of labeled data with a large amount of unlabeled data[12].

Motivation for semi-supervised algorithm is to overcome the problem of lack of annotated corpus and data scarcity problem. Semi-supervised usually starts with small amount of annotated corpus, large amount of un-annotated corpus and a small set initial hypothesis or classifiers. With each iteration, more annotations are generated and stored until a certain threshold occurs to stop the iterations[13]. The term “semi-supervised"(or “weakly supervised") is relatively recent. The main technique for SSL is called “bootstrapping" and involves a small degree of supervision, such as a set of seeds, for starting the learning process. For example, a system aimed at “disease names" might ask the user to provide a small number of example names. Then the system searches for sentences that contain these names and tries to identify some contextual clues common to the random number of examples, say five examples. Then, the system tries to find other instances of disease names that appear in similar contexts. The learning process is then reapplied to the newly found examples, so as to discover new relevant contexts. By repeating this process, a large number of disease names and a large number of contexts are then identified.

In bootstrapping algorithm, which is an iterative algorithm usually runs on a large collection of text documents. The algorithm has been studied extensively and employed in a number of applications. Important applications based on this algorithm include:

1. The extraction of the semantic lexicons, where the term semantic lexicon refers to a dictionary of words labeled with semantic categories, such as vehicles, animals, events, etc.
2. The recognition of named entities such as persons, organizations or locations.
3. The extraction of pair-wise named entities that share a certain relation, where the relation can be organization-hash-headquarters-in for organization and location pairs.

Fig 2 shows the steps in Bootstrapping. The system starts with a small number of seed examples, which are provided by the user. The system then finds occurrences of these examples in a large set of documents. By analyzing these occurrences, the system generates contextual extraction patterns (rules) and assigns confidence scores to the patterns. After this step, the system applies the extraction patterns to the documents and extracts new candidates. Based on some validation mechanism, the system assigns scores to the extracted candidates, and chooses the best ones to add to the seed set. Then the system starts over to perform many similar iterations, and at every iteration it learns more patterns and can extract more instances.

C. Unsupervised methods

A major problem with supervised setting is requirement of specifying large number of features. For learning good model, a robust set of features and large annotated corpus is needed. Many languages don't have large annotated corpus available at their disposal. To deal with lack of annotated text across domains and languages, unsupervised techniques for NER have been proposed. The typical approach in unsupervised learning is clustering. For example, one can try to gather named entities from clustered groups based on the similarity of context. There are other unsupervised methods too. Basically, the techniques rely on lexical resources on lexical patterns and on statistics computed on a large un-annotated corpus.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

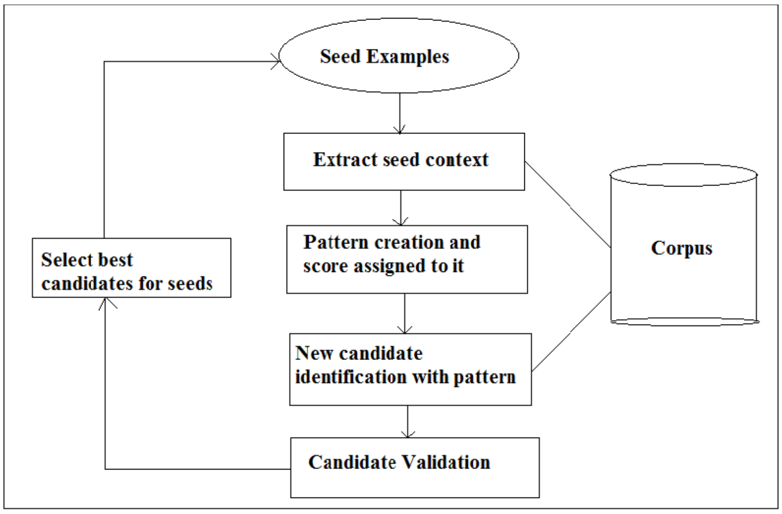

Fig 2. Steps in Boot-strapping

KNOWITALL is domain independent system that extracts information from the web in an unsupervised, open-ended manner. KNOWITALL uses 8 domain independent extraction patterns to generate candidate facts. The two main KNOWITALL modules are the Extractor and the Assessor. The Extractor creates a query from keywords in each rule, sends the query to a Web search engine, and applies the rule to extract information from the resulting Web pages. The Assessor computes a probability that each extraction is correct before adding the extraction to KNOWITALL's knowledge base. The Assessor bases its probability computation on search engine hit counts used to compute the mutual information between the extracted instance of a class and a set of automatically generated discriminator phrases associated with that class[14].

A Bootstrapping step creates extraction rules and discriminators for each predicate in the focus. KNOWITALL creates a list of search engine queries associated with the extraction rules, then executes the main loop. At the start of each loop, KNOWITALL selects queries, favoring predicates and rules that have been most productive in previous iterations of the main loop. The Extractor sends the selected queries to a search engine and extracts information from the resulting Web pages. The Assessor computes the probability that each extraction is correct and adds it to the knowledge base. This loop is repeated until all queries are exhausted or deemed too unproductive. KNOWITALL's running time increases linearly with the size and number of web pages it examines. KNOWITALL's PMI-based Assessor is effective at sorting extracted instances by their likelihood of being correct in order to achieve a reasonable precision/recall tradeoff.

V. Hybrid Approach

As discussed earlier, both rule based and learning based approaches have their own limitations. Combination of these approaches can be used to overcome individual's drawbacks and get improved performance. The hybrid approach integrates the rule-based approach with the ML-based approach in order to optimize the overall performance. In hybridization of NER, any machine learning approach/technique is combined with rules to increase the efficiency. In many cases, (e.g. for non English languages like Arabian) it may require to use grammar rules to get named entities. Here, only machine learning will not serve for the purpose and hybridization need to be used for good performance.

Recently many systems are build on this approach. Hybrid NER system for Hindi language has proposed[11]. In this system CRF approach is combined with Hindi grammar rules to enhance performance of NER system which is designed for Hindi language. Similarly a pipelined NER approach for Arabic language has been proposed [8] which also combined machine learning method with the Arabic grammar rules. Though the hybrid approach is most recent, it has been adopted widely due to its flexibility and good results. Languages having different syntax and semantics from English language are mostly benefited from this approach.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

VI. CONCLUSION

This paper surveyed almost all important approaches of NER in detail. These approaches are mainly categorizes in rule based and learning based. Recently hybrid approach become third category where it integrates previous approaches. Older NER systems were mostly based on rule based approach. Advantages and limitations of these approaches are discussed in this paper. Multiple methods under machine learning (supervised, semi-supervised and unsupervised) used by different application according to availability of annotated data available for training purpose. Future scenario for most NER system will be making use of hybrid approach as it combines the advantages of machine learning techniques with obtained efficiency from rule based methods.

REFERENCES

1. Sundheim, Beth M. “Overview of results of the MUC-6 evaluation.” Proceedings of a workshop on held at Vienna, Virginia: May 6-8, 1996. Association for Computational Linguistics, 1996.
2. Duan, Huanzhong, and Yan Zheng. “A study on features of the CRFs-based Chinese Named Entity Recognition." International Journal of Advanced Intelligence 3.2 (2011): 287-294.
3. Keretna, Sara, et al. “Classification ensemble to improve medical Named Entity Recognition." Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on. IEEE, 2014
4. Ratinov, Lev, and Dan Roth. “Design challenges and misconceptions in named entity recognition." Proceedings of the Thirteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2009.
5. Nadeau, David, and Satoshi Sekine. “A survey of named entity recognition and classification." Linguistic Investigations 30.1 (2007): 3-26.
6. Ponomareva, Natalia, et al. “Conditional random fields vs. hidden markov models in a biomedical named entity recognition task." Proc. of Int. Conf. Recent Advances in Natural Language Processing, RANLP. 2007.
7. Borthwick, Andrew. A maximum entropy approach to named entity recognition. Diss. New York University, 1999.
8. Oudah, Mai, and Khaled F. Shaalan. "A Pipeline Arabic Named Entity Recognition using a Hybrid Approach." COLING. 2012.
9. Dehghan, Azad, John A. Keane, and Goran Nenadic. “Challenges in Clinical Named Entity Recognition for Decision Support." Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on. IEEE, 2013.
10. Xing, Zhengzheng, Jian Pei, and Eamonn Keogh. “A brief survey on sequence classification." ACM SIGKDD Explorations Newsletter 12.1 (2010): 40-48.
11. 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).
12. Sutton, Charles, and Andrew McCallum. “An introduction to conditional random fields." Machine Learning 4.4 (2011): 267-373.
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
16. Marsh, E., and D. Perzanowski. “Muc-7 evaluation of information extraction technology: Overview of results." Seventh Message Understanding Conference (MUC-7). 1998.
17. Chiticariu, Laura, et al. “Domain adaptation of rule-based annotators for named-entity recognition tasks." Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2010.
18. Isozaki, Hideki, and Hideto Kazawa. “Efficient support vector classifiers for named entity recognition." Proceedings of the 19th international conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2002.
19. Bikel, Daniel M., Richard Schwartz, and Ralph M. Weischedel. “An algorithm that learns what's in a name." Machine learning 34.1-3 (1999): 211-231.
20. Finkel, Jenny Rose, Trond Grenager, and Christopher Manning. “Incorporating non-local information into information extraction systems by gibbs sampling.”Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2005.
21. Kazama, Jun'ichi, and Kentaro Torisawa. “A new perceptron algorithm for sequence labeling with non-local features." Proc. of EMNLP-CoNLL. Vol. 7. 2007.