Contents

Chapter 1: Introduction

1.1 Named Entities

The term “Named Entity”, now widely used in Natural Language Processing, was coined for the Sixth Message Understanding Conference (MUC-6)[GRI96].Broadly speaking, named entities are proper nouns. However, named entity tasks often include expressions for date and time, names of sports and adventure activities, terms for biological species and substances as named entities. MUC- 7[CHI98] classifies named entities into following categories and subcategories:

1. Entity (ENAMEX): person, organization, location
2. Time expression (TIMEX): date, time
3. Numeric expression (NUMEX): money, percent.

1.2 Named Entity Recognition and Classification (NER)

It was noticed that it is essential to recognize information units like names, including person, organization and location names, and numeric expressions including time, date, money and percent expressions for various Information Extraction and NLP tasks. Identifying references to these entities in text was recognized as one of the important sub-tasks of IE and was called “Named Entity Recognition and Classification (NER)”.

Though this sounds clear, special cases arise to require lengthy guidelines, *e.g.*, when is *The Times of India* an artifact, and when is it an organization? When is *White House* an organization, and when a location? Are branch offices of a bank an organization? Is a garment factory a location or an organization? Is a street name a location? Is a phone number a numeric expression or is it an address (location). Is *mid-morning* a time? In order to achieve human annotator consistency, guidelines with numerous special cases have been defined for the Seventh Message Understanding Conference, MUC-7 [CHI98].

Most research on NER systems has been structured as taking an unannotated block of text, for *e.g.*: “The delegation, which included the commander of the U .N. troops in Bosnia , Lt. Gen. Sir Michael Rose reached Sarajevo on 13th October .” and producing an annotated block of text- “The delegation, which included the commander of the <ORG> U .N. </ORG> troops in <LOC> Bosnia </LOC>, <PERS>Lt. Gen. Sir Michael Rose </PERS> reached <LOC> Sarajevo </LOC> on <TIME>13th October </TIME>”.

Both the boundaries of an expression and its label must be marked.

1.2 Applications of NER

NER finds application in most of the NLP applications. The following list mentions few of its applications. [WIK], [YOO07]

1) NER is very useful for search engines. NER helps in structuring textual information, and structured information helps in efficient indexing and retrieval of documents for search.
2) In the context of Cross-Lingual Information Access Retrieval (CLIR), given a query word, it is very important to find if it is a named entity or not. If a query word is a Named Entity, we need to transliterate a query word, rather than translating it.
3) The new generation of news aggregation platforms is powered by named entity recognition. A lot of information can be analyzed using named entities, like plotting the popularity of entities over time and generating geospatial heat maps However, the main improvement to traditional news aggregation brought by NEs is how they connect between people and things.
4) NER finds application in machine translation, as well. Usually, entities identified as Named Entities are transliterated as opposed to getting translated.

5) Before reading an article, if the reader could be shown the named entities, the user would be able to get a fair idea about the contents of the article.
6) Automatic indexing of Books: Most of the words indexed in the back index of a book are Named Entities.
7) Useful in Biomedical domain to identify Proteins, medicines, diseases, *etc*.
8) NE Tagger is usually a sub-task in most of the information extraction tasks because it adds structure to raw information.

The Cross-Lingual information access system also has a NER module. The CLIA pipeline is described in section 1.2.1

1.2.1 Introduction to Cross-Lingual Information Access (CLIA)

Cross Lingual information access (CLIA) is a mission mode project to be executed by a consortium of academic and research institutions and industry partners in India. The expected deliverables of the project are

- A user will be able to give a query in one Indian language
- The user will be able to access documents available in
  - the query language
  - Hindi (if the query language is not Hindi)
  - English

Users might not be familiar with the language of the documents retrieved. The CLIA system aims to remove this handicap. This requires additional processing which may be in the form of machine translation, transliteration, disambiguation of summaries and/or information extraction.

The overall architecture of the system is given in Figure 1.1.

Different modules in this architecture are:-

- Language Analyzer
  a. Tokenization
  b. Stop word Removal
  c. Stemming
- **NER**

- MWE
- Translation
    a. Translation
    b. Transliteration
- Query Disambiguation

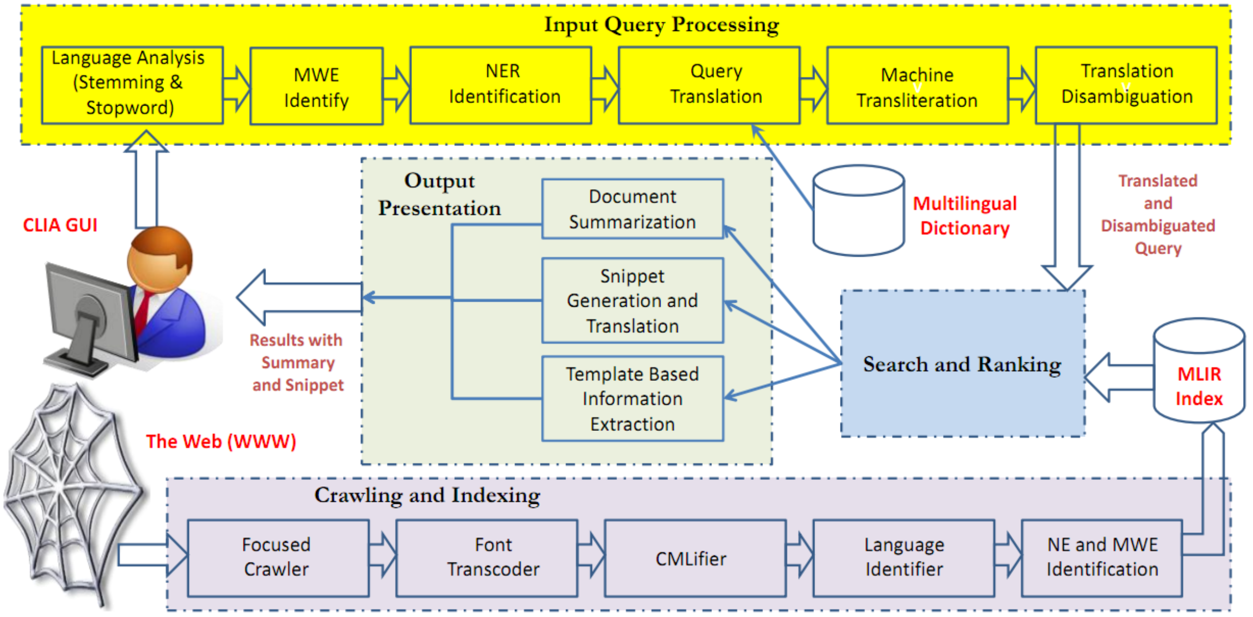


Figure 1.1: CLIA Architecture

1.3 Challenges in NER for Indian Languages

Accurate named entity recognition systems are now available for European languages especially English, and even for East Asian languages However, for South and South East Asian languages, the problem of NER is still far from being solved. Even though we can gain much insight from the methods used for English, there are many issues which make the nature of the problem different for Indian languages. For example, these languages do not have capitalization, which is a major feature used by NER systems for European languages. Another characteristic of these languages is that most of them use scripts of Brahmi origin, which have highly phonetic characteristics that could be utilized for multilingual NER.

Large gazetteers are not available for most of these languages. There is also the problem of lack of standardization and spelling variation. The number of frequently used words (common nouns) which can also be used as names (proper nouns) is very large for, unlike for European languages where a larger proportion of the first names are not used as common words. For example, ‘Smith’, ‘John’, ‘Thomas’ and ‘George’ *etc*. are almost always used as person names, but ‘Anand’, ‘Vijay’, ‘Kiran’ and even ‘Manmohan’ can be (more than often) used as common nouns. And the frequency with which they can be used as common nouns as against person names is more or less unpredictable.

Among other problems, one example is that of the various ways of representing abbreviations. Because of the alpha-syllabic nature of the Indian scripts, abbreviation can be expressed through a sequence of letters or syllables, but most importantly, there is a serious lack of labeled data for machine learning.

1.4 Roadmap

In Chapter 2, we explore and analyze various models developed for NER over the years. In Chapter 3, we describe the work done at IIT Bombay in the field of NER for Indian languages especially the CLGIN system.

Chapter 2: NER Survey

In this chapter we present our survey of all the major work done in NER over the past years. Sections 2.2, 2.3, 2.4 list out and illustrate various factors that are important in NER. Section 2.5 is about the use of machine learning methods in NER and section 2.6 lists out various features that have been used in the past for these learning methods. Section 2.7 illustrates various statistical measures and their correlation with named entities. A large number of examples have been taken from [LI07].

2.1 General Observations

The computational research aiming at automatically identifying named entities in texts forms a vast and heterogeneous pool of strategies, methods and representations. One of the first research papers in the field was presented by Lisa F. Rau (1991). Rau's paper describes a system to "extract and recognize company names". It relies on heuristics and handcrafted rules. From 1991 to 1995, the publication rate remained relatively low. It accelerated in 1996, with the first major event dedicated to the task: MUC-6. It never declined since then with steady research and numerous scientific events: HUB-4 [CHI98], MUC- 7 and MET-2 , IREX [SEK00], CONLL [TJ02][TJ03], ACE [DOD04] and HAREM [SAN06]. The IJCNLP 08 conference also had a workshop and shared task on Named Entity Recognition for South and South East Asian Languages. The Language Resources and Evaluation Conference (LREC) have also been staging workshops and main conference tracks on the topic since 2000.

2.2 Language factor

A good proportion of work in NER research is devoted to the study of English but a possibly larger proportion addresses language independence and multilingualism problems. German is well studied in CONLL-2003 and in earlier works. Similarly, Spanish and Dutch are strongly represented, boosted by a major devoted conference: CONLL-2002. Japanese has been studied in the MUC-6 conference. Chinese is studied in an abundant literature ([WAN92] H.-H., [CHE96] [YU98]) and so are French ([PET01], [POI03]), Greek [BOU00] and Italian ([BLK98], [CUC01]). Many other languages received some attention as well: Basque (C. Whitelaw & Patrick 2003), Bulgarian [SLV04], Catalan [CRR03], Cebuano [May03], Danish [BCK04] Hindi ([CCZ99], [MAY03]), Korean [WHI03], Polish [PIS04], Romanian [CCZ09], Russian [POP04] Swedish [KKK98] and Turkish [CCZ09]. Portuguese was examined by [PLM97]. Arabic [HNG05] has also started to receive a lot of attention in large-scale projects such as Global Autonomous Language Exploitation (GALE). There have been major contributions for Indian languages as well – Hindi [SA08], Bengali [EKB08] , Oriya [BIS10] and Telugu [SAS11].

2.3 Textual genre or domain factor

The factor of textual genre (journalistic, scientific, informal, *etc*.) and domain (gardening, sports, business, *etc*.) has not been extensively studied in the NER literature. Few studies are specifically devoted to various genres and domains. Maynard et al. [MYD01] designed a system for emails, scientific texts and religious texts. Minkov et al. [MIN05] created a system specifically designed for email documents. These experiments demonstrated that although any domain can be reasonably supported, porting a system to a new domain or textual genre remains a major challenge. Poibeau et al. [POI01], for instance, tested some systems on both the MUC-6 collection composed of newswire texts, and on a proprietary corpus made of manual translations of phone conversations and technical emails. They reported a drop in performance for every system (some 20% to 40% of precision and recall). The Named Entity Recognition module being developed for the CLIA system at IIT Bombay is focused mainly on the tourism domain.

2.4 Entity type factor

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 referent. 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 kind terms like biological species and substances. There is a general agreement in the NER community 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*.). It is arguable that the NE definition is loosened in such cases for practical reasons.

Overall, the most studied types are three specializations of “proper names”: names of “persons”, “locations” and “organizations”. These types are collectively known as “enamex” since the MUC-6 competition. The type “location” can in turn be divided into multiple subtypes of “fine grained locations”: city, state, country, *etc*. ([FL01], [LEE05]). Similarly, “fine-grained person” sub-categories like “politician” and “entertainer” appear in the work of [FL01] and [HOV02]. The type “person” is quite common and used at least once in an original way by [BOD00] who combines it with other cues for extracting medication and disease names (*e.g.*, “Parkinson disease”). In the ACE program, the type “facility” subsumes entities of the types “location” and “organization”. The type “GPE” is used to represent a location which has a government, such as a city or a country.

The type “miscellaneous” is used in the CONLL conferences and includes proper names falling outside the classic “enamex”. The class is also sometimes augmented with the type “product” [BCK04]. The “timex” (another term coined in MUC) types “date” and “time” and the “numex” types “money” and “percent” are also quite predominant in the literature.. Finally, marginal types are sometime handled for specific needs: “film” and “scientist” ([ETZ05]), “email address” and “phone number” ([WTT99], [MYD01]), “research area” and “project name” [ZHU05]“book title" (S. Brin 1998, [WTT99]), "job title" [COH04] and "brand" [BCK04]. A recent interest in bioinformatics, and the availability of the GENIA corpus [OHT02] led to many studies dedicated to types such as "protein", "DNA", "RNA", "cell line" and "cell type" (*e.g.*, [SH03], [STT04]) as well as studies targeted to "protein" recognition only (Y. Tsuruoka & Tsujii 2003). Related work also includes "drug" [RIN00] and "chemical" [NAR03] names.

We now describe the NER tagset used for CLIA. The Named entity hierarchy is divided into three major classes; Entity Name, Time and Numerical expressions. The Name hierarchy has eleven attributes. Numeral Expression and time have four and three attributes respectively.

There are eleven types of entities in Name as given below.

1. **Person**: Person entities are limited to humans. A person may be a single individual or a group.

2. **Organization**: Organization entities are limited to corporations, agencies, and other groups of people defined by an established organizational structure.

3. **Location:** Location entities are limited to geographical entities such as geographical areas and landmasses, bodies of water, and geological formations.

4. **Facilities:** Facility entities are limited to buildings and other permanent man-made structures and real estate improvements.

5. **Locomotives:** A locomotive entity is a physical device primarily designed to move an object from one location to another, by (for example) carrying, pulling, or pushing the transported object. Vehicle entities may or may not have their own power source.

6. **Artifacts:** Artifact entities are objects or things, which are produced or shaped by human craft, such as tools, weapons/ammunition, art paintings, clothes, ornaments, medicines.

7. **Entertainment:** Entertainment entities denote activities, which are diverting and hold human attention or interest, giving pleasure, happiness, amusement especially performance of some kind such as dance, music, sports, events.

8. **Cuisines:** This entity refers to various type of food, prepared in different manners such as Chinese food, South-Indian, North-Indian foods.

9. **Organisms**: Organism entities are living things and have the ability to act or function independently such as humans, viruses, bacteria *etc*. Here we have not taken into consideration plants, those have been classified separately.

10. **Plants:** These entities are living things having photosynthetic, eukaryotic, multicellular organisms of the kingdom Plantae, containing chloroplasts, having cellulose cell walls, and lacking the power of locomotion.

11. **Disease:** This entity refers to the state of a disordered or incorrectly functioning organ, part, structure, or system of the body resulting from the effect of genetic or developmental errors, infection, poisons, nutritional deficiency or imbalance, toxicity, or unfavorable environmental factors; illness; sickness; ailment such as fever, cancer *etc*.

2.5 Learning methods

The ability to recognize previously unknown entities is an essential part of NER systems. Such ability hinges upon recognition and classification rules triggered by distinctive features associated with positive and negative examples. While early studies were mostly based on handcrafted rules, most recent ones use supervised machine learning (SL) as a way to automatically induce rule-based systems or sequence labeling algorithms starting from a collection of training examples. This is evidenced, in the research community, by the fact that five systems out of eight were rule-based in the MUC-7 competition while sixteen systems were presented at CONLL-2003, a forum devoted to learning techniques. When training examples are not available, handcrafted rules remain the preferred technique, as shown in [SEK04] who developed a NER system for 200 entity types.

The idea of supervised learning is to study the features of positive and negative examples of NE over a large collection of annotated documents and design rules that capture instances of a given type. Section 2.5.1 explains SL approaches in more details. The main shortcoming of SL is the requirement of a large annotated corpus. The unavailability of such resources and the prohibitive cost of creating them lead to two alternative learning methods: semi-supervised learning (SSL) and unsupervised learning (UL). These techniques are presented in section 2.5.2 and 2.5.3 respectively.

The performance of a Named Entity Recognition system is measure in terms of the following three parameters: Precision (P), Recall (R) and F1- value where,

$$P = \frac{Number\ of\ correct\ tags\ assigned}{Total\ number\ of\ tags\ assigned}$$

$$R = \frac{Number\ of\ correct\ tags\ assigned}{Total\ number\ of\ tags\ in\ the\ annotated\ test\ corpus}$$

$$F1 = \frac{2RP}{R + P}$$

2.5.1 Supervised learning

The current dominant technique for addressing the NER problem is supervised learning. SL techniques include Hidden Markov Models (HMM) [BIK97], Decision Trees [SEK98], Maximum Entropy Models (ME) [BOR98], Support Vector Machines (SVM) [ASA03], and Conditional Random Fields (CRF) [MCM03]. These are all variants of the SL approach that typically consist of a system that reads a large annotated corpus, memorizes lists of entities, and creates disambiguation rules based on discriminative features.

A baseline SL method that is often proposed consists of tagging words of a test corpus when they are annotated as entities in the training corpus. The performance of the baseline system depends on the vocabulary transfer, which is the proportion of words, without repetitions, appearing in both training and testing corpus. Palmer et al. [PLM97] calculated the vocabulary transfer on the MUC-6 training data. They report a transfer of 21%, with as much as 42% of location names being repeated but only 17% of organizations and 13% of person names. Vocabulary transfer is a good indicator of the recall (number of entities identified over the total number of entities) of the baseline system but is a pessimistic measure since some entities are frequently repeated in documents. Mikheev et al. [MKV99] precisely calculated the recall of the baseline system on the MUC-7 corpus. They report a recall of 76% for locations, 49% for organizations and 26% for persons with precision ranging from 70% to 90%. [WHI03] report consistent results on MUC-7 for the aggregated enamex class. For the three enamex types together, the precision of recognition is 76% and the recall is 48%.

2.5.2 Semi-supervised learning

The main technique for Semi-supervised Learning (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 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 will eventually be gathered. Experiments in semi-supervised NER [NAD06] report performances that rival baseline supervised approaches. Here are some examples of SSL approaches.

Brin et al. [BRN98] uses lexical features implemented by regular expressions in order to generate lists of book titles paired with book authors. It starts with seed examples such as {Isaac Asimov, The Robots of Dawn} and use some fixed lexical control rules such as the following regular expression [A-Z][A-Za-z .,&]5,30[A-Za-z.] used to describe a title. The main idea of his algorithm, however, is that many web sites conform to a reasonably uniform format across the site. When a given web site is found to contain seed examples, new pairs can often be identified using simple constraints such as the presence of identical text before, between or after the elements of an interesting pair. For example, the passage “The Robots of Dawn, by Isaac Asimov (Paperback)” would allow finding, on the same web site,” The Ants, by Bernard Werber (Paperback)”.

Collins et al.[COL99] parse a complete corpus in search of candidate NE patterns. A pattern is, for instance, a proper name (as identified by a part-of-speech tagger) 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. Following the steps above, contextual rules can be used to find further spelling rules, and so on.

E. Riloff and Jones [RIL99] introduce mutual bootstrapping that consists of growing a set of entities and a set of contexts in turn. Instead of working with predefined candidate NE’s (found using a fixed syntactic construct), they start with a handful of seed entity examples of a given type (e.g., Bolivia, Guatemala, Honduras are entities of type country) and accumulate all patterns found around these seeds in a large corpus. Contexts (*e.g.*, offices in X, facilities in X, …) are ranked and used to find new examples.

Riloff and Jones note that the performance of that algorithm can deteriorate rapidly when noise is introduced in the entity list or pattern list. While they report relatively low precision and recall in their experiments, their work proved to be highly influential.

Cucchiarelli et al.[CUC01] use syntactic relations (*e.g.*, subject-object) to discover more accurate contextual evidence around the entities. Again, this is a variant of E. Riloff and Jones mutual bootstrapping (1999). Interestingly, instead of using human generated seeds, they rely on existing NER systems (called early NE classifier) for initial NE examples.

Pasca et al. [PAS06] are also using techniques inspired by mutual bootstrapping. However, they innovate through the use of [LIN98] distributional similarity to generate synonyms – or, more generally, words which are members of the same semantic class – allowing pattern generalization. For instance, for the pattern X was born in November, Lin's synonyms for November are {March, October, April, Mar, Aug., February, Jul, Nov., ...} thus allowing the induction of new patterns such as X was born in March. One of the contributions of [PAS06] is to apply the technique to very large corpora (100 million web documents) and demonstrate that starting from a seed of 10 examples facts (defined as entities of type person paired with entities of type year - standing for the person year of birth) it is possible to generate one million facts with a precision of about 88%. The problem of unlabeled data selection is addressed by [HEN06]. They show how an existing NE classifier can be improved using bootstrapping methods. The main lesson they report is that relying upon large collection of documents is not sufficient by itself. Selection of documents using information retrieval-like relevance measures and selection of specific contexts that are rich in proper names and coreferences bring the best results in their experiments.

2.5.3 Unsupervised learning

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 (*e.g.*, WordNet), on lexical patterns and on statistics computed on a large unannotated corpus. Here are some examples.

Alfonseca et al.[ALF02] study 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*.). The 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.

In [EV03], the method for identification of hyponyms/hypernyms described in the work of [HRT92] is applied in order to identify potential hypernyms of sequences of capitalized words appearing in a document. For instance, when X is a capitalized sequence, the query “such as X”, is searched on the web and, in the retrieved documents, the noun that immediately precede the query can be chosen as the hypernym of X. Similarly, in [CIM05], Hearst patterns are used but this time, the feature consists of counting the number of occurrences of passages like: “city such as”, “organization such as”, *etc*.

Sekine et al. [SEK04] used an observation that named entities often appear synchronously in several news articles, whereas common nouns do not. They found a strong correlation between being a named entity and appearing punctually (in time) and simultaneously in multiple news sources. This technique allows identifying rare named entities in an unsupervised manner and can be useful in combination with other NER methods.

In [ETZ05], Pointwise Mutual Information and Information Retrieval (PMI-IR) is used as a feature to assess that a named entity can be classified under a given type. PMI-IR, developed by [TUR01], measures the dependence between two expressions using web queries. A high PMI-IR means that expressions tend to co-occur. Etzioni et al. [ETZ05] create features for each candidate entity (*e.g.*, London) and a large number of automatically generated discriminator phrases like “is a city”, “nation of”, *etc.*

2.6 Feature space for Named Entity Recognition

Features are descriptors or characteristic attributes of words designed for algorithmic consumption. An example of a feature is a Boolean variable with the value *true* if a word is capitalized and *false* otherwise. Feature vector representation is an abstraction over text where typically each word is represented by one or many Boolean, numeric and nominal values. For example, a hypothetical NER system may represent each word of a text with 3 attributes:

1) A Boolean attribute with the value *true* if the word is capitalized and *false* otherwise;
2) A numeric attribute corresponding to the length, in characters, of the word;
3) A nominal attribute corresponding to the lowercased version of the word.

In this scenario, the sentence “The president of Apple eats an apple.” excluding the punctuation, would be represented by the following feature vectors:

<true, 3, “the”>, <false, 9, “president”>, <false, 2, “of”>, <true, 5, “apple”>, <false, 4, “eats”>, <false, 2, “an”>, <false, 5, “apple”>

Usually, the NER problem is resolved by applying a rule system over the features. For instance, a system might have two rules, a recognition rule: “capitalized words are candidate entities” and a classification rule: “the type of candidate entities of length greater than 3 words is organization”. These rules work well for the sentence above. However, real systems tend to be much more complex and their rules are often created by automatic learning techniques.

In this section, we present the features most often used for the recognition and classification of named entities. We organize them in 2 categories: Word-level features and List lookup features

2.6.1 Word-level features

Word-level features are related to the character makeup of words. They specifically describe word case, punctuation, numerical value and special characters. Table 2.1 lists subcategories of word-level features.

2.6.1.1 Digit pattern

Digits can express a wide range of useful information such as dates, percentages, intervals, identifiers, *etc*. Special attention must be given to some particular patterns of digits. For example, two-digit and four-digit numbers can stand for years [BIK97] and when followed by an “s”, they can stand for a decade; one and two digits may stand for a day or a month [YU98].

Features Examples	Examples
Case	• Starts with a capital letter • Word is all uppercased • The word is mixed case (*e.g.*, ProSys, eBay)
Punctuation	• Ends with period, has internal period (*e.g.*, St., I.B.M.) • Internal apostrophe, hyphen or ampersand (*e.g.*, O’Connor)
Digit	• Digit pattern (*see section 1.5.1.1*) • Cardinal and Ordinal • Roman number • Word with digits (*e.g.*, W3C, 3M)
Character	• Possessive mark, first person pronoun • Greek letters
Morphology	• Prefix, suffix, singular version, stem • Common ending (*see section 1.5.1.2*)
Part-of-speech	• proper name, verb, noun, foreign word
Function	• Alpha, non-alpha, n-gram (*see section 1.5.1.3*) • lowercase, uppercase version • pattern, summarized pattern (*see section 1.5.1.4*) • token length, phrase length

Table 2.1:Word-level features for NER

2.6.1.2 Common word ending

Morphological features are essentially related to words affixes and roots. For instance, a system may learn that a human profession often ends in “ist” (*journalist*, *cyclist*) or that nationality and languages often ends in “ish” and “an” (*Spanish, Danish, Romanian*). Another example of common word ending is organization names that often end in “ex”, “tech”, and “soft” [BCK04].

2.6.1.3 Functions over words

Features can be extracted by applying functions over words. An example is given by M. [COL99] who create a feature by isolating the non-alphabetic characters of a word (*e.g.*, nonalpha(A.T.&T.) = ..&.) Another example is given by [PAT02] who use character n-grams as features.

2.6.1.4 Patterns and summarized patterns

Pattern features were introduced by [COL02] and then used by others (W. [COH04] and [SET04]). Their role is to map words onto a small set of patterns over character types. For instance, a pattern feature might map all uppercase letters to “A”, all lowercase letters to “a”, all digits to “0” and all punctuation to “-”:

x = "G.M.": GetPattern(x) = "A-A-"

x = "Machine-223": GetPattern(x) = "Aaaaaaa-000"

The summarized pattern feature is a condensed form of the above in which consecutive character types are not repeated in the mapped string. For instance, the preceding examples become:

x = "G.M.": GetSummarizedPattern(x) = "A-A-"

x = "Machine-223": GetSummarizedPattern(x) = "Aa-0"

2.6.2 List lookup features

Lists are the privileged features in NER. The terms “gazetteer”, “lexicon” and “dictionary” are often used interchangeably with the term “list”. List inclusion is a way to express the relation “is a” (*e.g.*, *Paris is a city*). It may appear obvious that if a word (*Paris*) is an element of a list of cities, then the probability of this word to be city, in a given text, is high.

2.6.2.1 General dictionary

Common nouns listed in a dictionary are useful, for instance, in the disambiguation of capitalized words in ambiguous positions (*e.g.*, sentence beginning). Mikheev et al. [MKV99] reports that from 2677 words in ambiguous position in a given corpus, a general dictionary lookup allows identifying 1841 common nouns out of 1851 (99.4%) while only discarding 171 named entities out of 826 (20.7%). In other words, 20.7% of named entities are ambiguous with common nouns, in that corpus.

Features	Examples
General list	- General dictionary (see section 2.2.1) - Stop words (function words) - Capitalized nouns (*e.g.*, January, Monday) - Common abbreviations
List of entities	- Organization, government, airline, educational - First name, last name, celebrity - Astral body, continent, country, state, city
List of entity cues	- Typical words in organization (see 2.2.2) - Person title, name prefix, post-nominal letters - Location typical word, cardinal point

Table 2.2: List lookup features for NER

2.6.2.2 Words that are typical of organization names

Many authors propose to recognize organizations by identifying words that are frequently used in their names. For instance, knowing that “associates” is frequently used in organization names could lead to the recognition of “Computer Associates” and “BioMedia Associates” ([MCD93], [GAI95]). The same rule applies to frequent first words (“Indian”, “General”) of an organization [RAU91] Some authors also exploit the fact that organizations often include the name of a person ([WOL95], [RAV96]) as in “Alfred P. Sloan Foundation”. Similarly, geographic names can be good indicators of an organization name [WOL95] as in “China Telecom”. Organization designators such as “inc” and “corp” [RAU91] are also useful features.

2.6.2.3 List lookup techniques

Most approaches implicitly require candidate words to exactly match at least one element of a pre-existing list. However, we may want to allow some flexibility in the match conditions. At least three alternate lookup strategies are used in the NER field. First, words can be stemmed (stripping off both inflectional and derivational suffixes) or lemmatized (normalizing for inflections only) before they are matched [COA92]. For instance, if a list of cue words contains “technology”, the inflected form “technologies” will be considered as a successful match. For some languages [JAN02], diacritics can be replaced by their canonical equivalent (*e.g.*, ‘é’ replaced by ‘e’). Second, candidate words can be “fuzzy-matched” against the reference list using some kind of thresholded edit-distance ([TSU03]) or Jaro-Winkler [COH04]This allows capturing small lexical variations in words that are not necessarily derivational or inflectional. For instance, *Frederick* could match *Frederik* because the edit-distance between the two words is very small (suppression of just one character, the ‘c’). Jaro-Winkler’s metric was specifically designed to match proper names following the observation that the first letters tend to be correct while name ending often varies. Third, the reference list can be accessed using the Soundex algorithm [RAG04] which normalizes candidate words to their respective Soundex codes. This code is a combination of the first letter of a word plus a three digit code that represents its phonetic sound. Hence, similar sounding names like *Lewinskey* (soundex *= l520*) and *Lewinsky* (soundex *= l520*) are equivalent in respect to their Soundex code.

2.7 Corpus Statistics and NER

We look at various statistics which take into account the distribution of words across the entire corpus. Though the use of such statistics for NER has not been widely explored, we look at a few instances where they have been put to use.

2.7.1 Informativeness measures and statistics

It has been found that named entities are highly relevant to the topic of a document [CLI99]. Using measures or scores which give an indication of how topic-oriented or "informative" each word in a corpus is we can identify the named entities in the individual documents of the corpus. It is well known that informative words have "peaked" or "heavy-tailed" frequency distributions [CHU95]. Many scores including Inverse Document Frequency (IDF) [JON73], Residual IDF [CHU95], x_I [BO74], the z-measure [HA75] have been introduced to measure informativeness of words. The use of such measures is effective only when the corpus is sufficiently large. We now discuss these measures in detail.

2.7.1.1 Inverse Document Frequency (IDF)

Inverse document frequency (IDF) is an informativeness score. The principle behind the IDF measure is that the lesser number of documents a word occurs in, the greater is the chance that it is highly-relevant to those documents and greater is the information-content of the word. Specifically, the IDF score for a word, w, is

IDF= $-\log D_w/D$

where,

D_w = Number of documents the word w occurs in

D = Total number of documents in the corpus

The IDF score has been used to assign weights to words for information retrieval. It has also been used for text classification. Though IDF is important for various applications, when used in isolation it is a weak indicator of named entities. Thus we need additional scores other than IDF to detect named entities.

2.7.1.2 Residual Inverse Document Frequency (Residual IDF)

The concept of Expected IDF based on the frequency of words in the corpus was introduced by [CHU95]. To calculate the Expected IDF, each document was assumed to be a “bag of words” with no internal structure. The words were randomly generated by a Poisson process. They calculated the Expected IDF (E-IDF) as

$$\text{E-IDF} = -\log_2(1-e^{\theta})$$

where,

$\theta = f_w/D$

D = Total number of documents in the corpus

f_w = Total number of times the word w occurs in the corpus

It was noted that for nearly all the words E-IDF scores greater than IDF scores implying that words do not occur randomly across documents and documents have a structure. Residual IDF was introduced which is the difference between the observed IDF and the IDF that would be expected:

$$\text{Residual IDF} = \text{E-IDF} - \text{IDF}$$

Highly-relevant words like named entities are clustered into a few documents and hence, are expected to have higher Residual IDF scores as compared to less relevant words. [REN06] and [GUP10] experimented with various measures to detect named entities and found out that Residual IDF is the best individual score for detecting named entities.

Here’s an example to show how Residual IDF is more effective than IDF for detecting high-information content words and hence, named entities:

Consider two words- ‘boycott’ and ‘somewhat’. They both occurred approximately thousand times in a corpus of news articles. Hence, according to IDF measure, both have equal information content. However, clearly ‘boycott should have much more information –content as compared to ‘somewhat’. A good keyword, like ‘boycott’, picks out a very specific set of documents. The problem with somewhat is that it behaves almost like chance (Poisson). Under a Poisson, the 1013 instances of ‘somewhat’ should be found in approximately 1007 documents when we consider a Poisson distribution. In fact, ‘somewhat’ was found in 979 documents, only a little less than what would have been expected by chance. Good keywords tend to bunch up into many fewer documents, ‘boycott’, for example, bunch up into only 676 documents, much less than chance 1003 documents. Almost all words are more "interesting" in this sense than Poisson, but good keywords like boycott are a lot more interesting than Poisson and have high residual IDF measure, and others like ‘somewhat’ are only a little more interesting than Poisson and have low Residual IDF.

2.7.1.3 x^I measure

Harter et al. [HA75] introduced the x^I measure for a word w,

$x^I (w) = f_w - d_w$,

Where f_w is the frequency of word w and d_w is the document frequency of word w (number of documents in which w occurs). Informative words tend to exhibit “peaked” distributions with most occurrences coming in a handful of documents. This score makes sense at the intuitive level since for two words with the same frequency; the one that is more concentrated will have the higher score. However, this score has a bias toward frequent words, which tend to be less informative.

2.7.1.4 z measure

Harter et al. [HA75] noted that frequency statistics of informative or “specialty” words tend to fit poorly to a Poisson distribution. He suggested that informative words may be identified by observing their fit to a mixture of 2 Poissons ("2-Poisson") model; he introduced the z-measure as a criterion for identifying informative words. The z-measure, introduced earlier by [3], is a general measure between two distributions. It computes the difference between means divided by square-root of the summed variances:

$$z = \frac{\mu_1 - \mu_2}{\sqrt{\sigma_1^2 + \sigma_2^2}}$$

Harter found that this measure could be used to identify informative words for keyword indexing.

2.7.1.5 Gain

Pappini et al. [PAP01] derives the gain for a word w as

$$Gain(w) = \frac{d_w}{d}\left(\frac{d_w}{d} - 1 - log\left(\frac{d_w}{d}\right)\right)$$

where d_w is the document frequency of word w and D is the total number of documents. Extremely rare and extremely common words have low gain. Medium-frequency words have higher gain. A weakness of this measure is that it relies solely on document frequency—it does not take account for "peaked-ness" of a word's frequency distribution.

2.7.2 Application to NER

Most of the current approaches to NEI/NER do not use global distributional characteristics of words (*e.g.*, Information Content, Term Co-occurrence statistics, *etc.*) when a large corpus is under consideration. Some interesting feature functions were proposed by [SIL04] for multi-word units that can be thresholded using corpus statistics. However, those feature functions were based on capitalizations in words and hence, is not applicable for a large number of languages. Rennie et al. [REN05] introduced a new information measure and along with other scores described in previous sections used it for NE detection in informal communication (emails and bulletin boards). They used this approach for un-capitalized and ungrammatical English text, like bulletin boards where spellings and POS tags are not correct. They concluded that when used in isolation, Residual IDF is the best measure to detect named entities. Gupta et al. [GUP10] used global characteristics like information content, term co-occurrence statistics, *etc*. along with language cues to build features for a MEMM-based NER system for Hindi. They too concluded that Residual IDF is the best statistical measure for detecting named entities. Their work is described in detail in the Chapter 3.

Chapter 3: The CLGIN system

This chapter describes past work done at IIT Bombay in relation to NER. In particular, we describe the CLGIN[SH10] system which was a supervised system for NER. Section 3.1 describes the Statistical MEMM based supervised NER system. Section 3.2 gives a description of the CLGIN system. Section 3.3 describes experiments related to experiments in unsupervised NER.

3.1 MEMM Based NER System for Hindi

This section describes MEMM based system for Hindi NER. The system achieved an accuracy of 73%.

3.1.1 TagSet

The following tagset was used

Entity Type	Entity Tag
Person	NEP
Location	NEL
Organisation	NEO
Number	NEN
Measure	NEM
Time	NETI

Table 3.1: Tagset for MEMM system

3.1.2 Features Used

In this section, we will describe the features used.

1) *PER dict* : This feature is turned ON, when the current word exists in the gazetteer of person names and the POS tag is NNP.

2) *PER initials prev* : Checks if the previous word is one of possible initials. *E.g*. If *“shri", “dr.", etc*. precedes the given word whose tag is NNP.

3) *PER initials* : Checks if the current word is one of possible initials. *E.g*. If *“shri", “dr.", etc*. and a NNP follows this word.

4) *Initials* : Checks the previous word, if it is one of the possible initials like *“rashtrapati* ", *“vaigyanik*", then this feature is turned ON.

5) *PER nextContext* : This keeps track of the words that occur just next to the person words. All the NNPs are bypassed and then the system checks the next word following the NNPs.

6) *PER nextContext2* : Similar to the previous feature, but it takes care of next two words in context.

7) *LOC dict* : Similar to PER dict.

8) *LOC prev add* : Checks if the current word is a valid previous context for a location name and if the next word is an NNP (*E.g*. words like *“uttar ", “dakshin*", *etc*.). A list of these words is maintained.

9) *LOC next add* : Checks if the current word is a valid next context for a location name and if the previous word is an NNP (*E.g*. words like *“nadi", “tapu*", *etc*.). A list of these words is maintained.

10) *LOC suffix* : Checks if the suffix of the current word matches with common suffixes of Location names (*E.g*. *“garh", “pur*", *etc*.)

11) *LOC nextcontext1* : This keeps track of the words that occur just next to the location names. We first bypass all the list of NNPs and then check the next word following the NNPs.

12) *LOC nextcontext2* : Similar to LOC nextcontext1 feature. Context is of size 2.

13) LOC prevcontext1 and LOC *prevcontext2*: Similar to previous two features .

14) Features for organization names were similar. This included : *ORG dict, ORG next add, ORG next2 add, ORG prev2 add, ORG nextcontext1, ORG nextcontext2, ORG prevcontext1* and *ORG prevcontext2*- Context words: Previous and Next words.

15) *POS tag*.

16) *Lexicon*: This identifies if the word is present in lexicon or not. The viterbi implementation itself takes care of the previous output tag. For each tag, we generate the probability of reaching that tag from possible previous tags. At each step, for each label, we select the path which gives the highest probability of reaching the current state. For *"Number", "Measure"* and *"Time",* a system similar to the existing rule based system [Gup08] is used

3.1.3 Results

The following table contains accuracy figures for the new system.

Entity Tag	Precision	Recall	F-Measure
NEO	0.6555	0.3103	0.4212
NEP	0.8760	0.9471	0.9101
NEL	0.8280	0.7631	0.7941
NETI	0.6615	0.5119	0.5772
NEM	0.6682	0.8981	0.7663
NEN	0.6002	0.9059	0.7220
O	0.9938	0.9955	0.9946

Fig 3.2: Accuracy figures for CLIGN system

The overall accuracy is: 72.99% (Precision: 70.09%, Recall: 79.68%) for the current system and it was 61.99% (Precision: 69.43% , Recall: 59.26%) for the original system. [Roy08]

As, the figures show, there is a significant improvement in the accuracy of detecting organization names.

3.1.4 Using Foreign Language Word Information for NER

It was found that "Foreign Language Word" information is a useful feature for identifying organization tags. The accuracy for *"Organization"* tags improved from 42.12% to 45.9% which shows a 9% increase in accuracy for *"Organization"* names. The impact on overall accuracy of

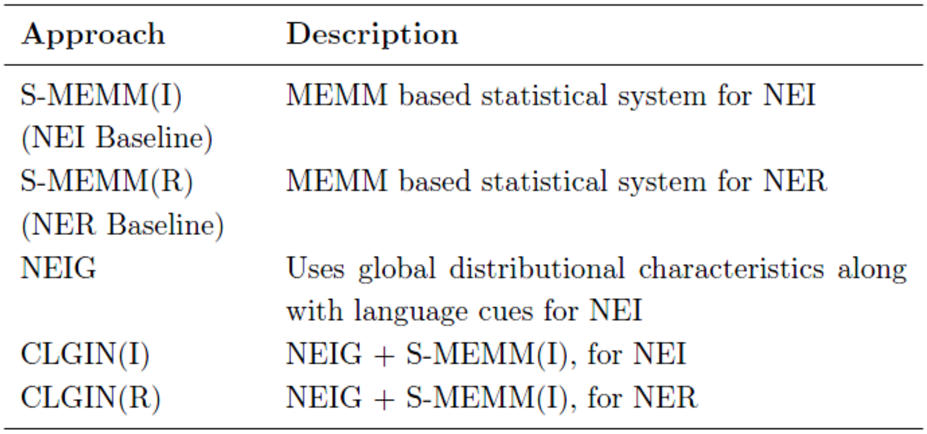
NER was not much. Foreign word feature simply checked that a word exists in this list or not. CFILT's statistical “Foreign Word Identifier" module was used to identify the foreign-language words. Including the new feature slightly increased the performance for NEO tags.

3.2 Combining Global and Local Characteristics for NEI and NER

For Indian languages, it is hard to identify named entities due to lack of capitalized letters in proper nouns. Many approaches based on MEMM [SSM08], CRFs [LM03] and hybrid models have been tried for Hindi Named Entity Recognition. These approaches use only the local context around the target word (context words, suffix information, POS tags, *etc*.) and gazetteers. Many applications need named entity identification in large corpora. When a large corpora need to be tagged, one can use the global characteristics of the words along with language dependent heuristics to identify the named entities. States of art -methods do not take advantage of these characteristics. Also, the performance of existing NER/NEI systems degrades substantially when the training and test corpus are from different domain or different genre.

A new approach-Combined Local and Global Information for Named Entity Recognition (CLGIN(R)) which combines the global characteristics with the local context for Hindi Named Entity Recognition was developed. The approach comprises of two steps:

1) Named Entity Identification using Global Information (NEIG) which uses the global distributional characteristics along with the language cues to identify NEs and
2) Combining the tagging from step 1 with the MEMM based statistical system.



Approach	Description
S-MEMM(I) (NEI Baseline)	MEMM based statistical system for NEI
S-MEMM(R) (NER Baseline)	MEMM based statistical system for NER
NEIG	Uses global distributional characteristics along with language cues for NEI
CLGIN(I)	NEIG + S-MEMM(I), for NEI
CLGIN(R)	NEIG + S-MEMM(I), for NER

Table 3.3: Summary of Approaches

3.2.1 CLGIN Approach

This section describes the CLGIN in approach. It combines the global information from the corpus with the local context. Figure 3.1 gives the block diagram of the system. This approach involves two steps:

1) Using NEIG to create a list of probable NEs using the whole corpus
2) Adding the tagging from step 1 as a feature in SMEMM. Output thus obtained from the MEMM system is the final output of the CLGIN approach.

The creation of list in step 1, involves the following sub steps:

1) A list of all words which appeared as a noun at least once in the corpus and which are not in the stop list is extracted.
2) The list is ordered on the basis of the information score derived using the whole corpus.
3) Words above the threshold (set during training using the development set) are selected as NEs.
4) Heuristics are applied for pruning and augmenting the ranked NE list.

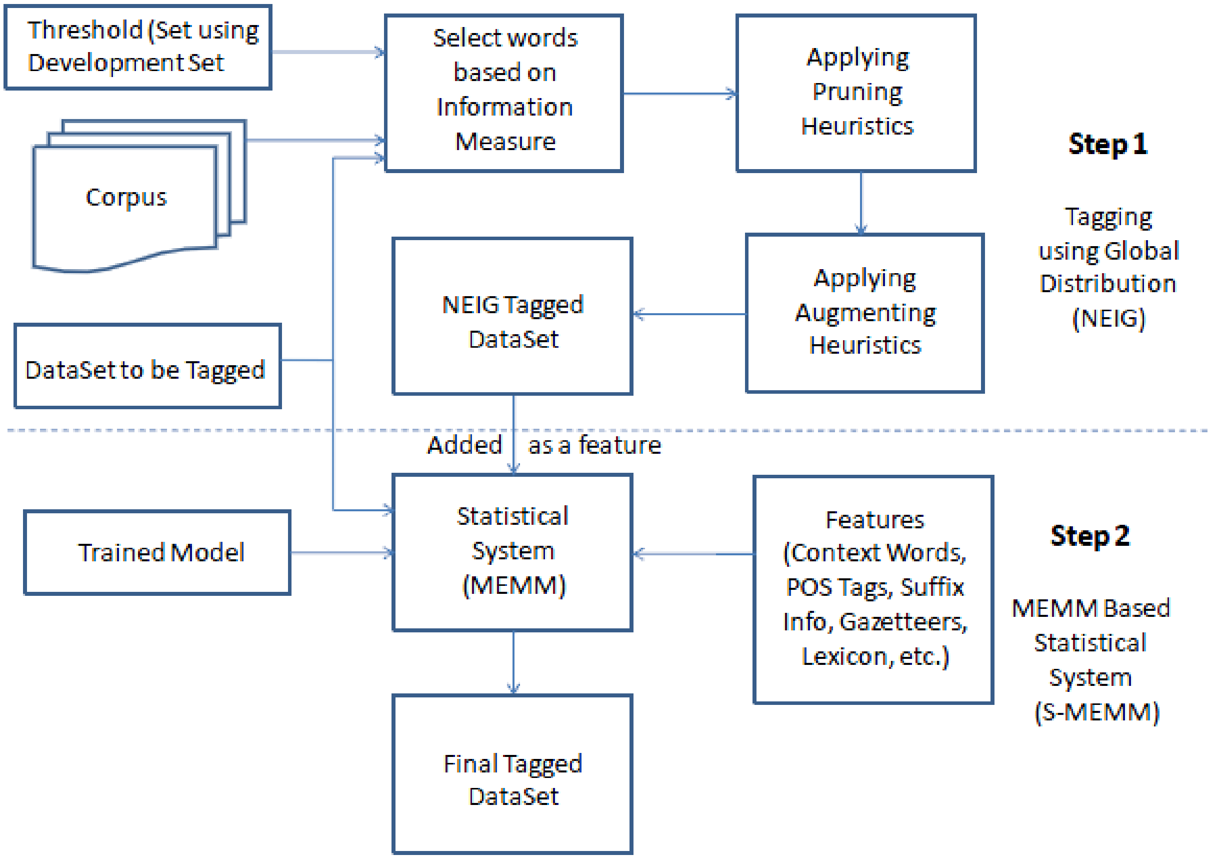

Threshold (Set using Development Set
Corpus
Select words based on Information Measure
Applying Pruning Heuristics
Step 1
Tagging using Global Distribution (NEIG)
NEIG Tagged DataSet
Applying Augmenting Heuristics
DataSet to be Tagged
Added as a feature
Trained Model
Statistical System (MEMM)
Features (Context Words, POS Tags, Suffix Info, Gazetteers, Lexicon, etc.)
Step 2
MEMM Based Statistical System (S-MEMM)
Final Tagged DataSet

Fig 3.1: Flowchart for CLGIN system[SH10]

3.2.2 Tagging using Global Distribution (NEIG)

This section describes in detail the processes involved in Step 1 (fig 3.1)

3.2.2.1 Information Measure/Score

NEs are highly relevant words in a document [CCR02] and are expected to have high information content [RJ05]. In this step, top few words with high information score are selected as NEs (threshold is set using a development set). Various information scores (IDF (Inverse Document Frequency) [Jon72], Residual IDF [CG95], x^{I} - measure [BS74], Gain [Pap01]) were compared. Of all the measures, Residual IDF performed best and was used to generate the ranked list of words which were expected to be NEs using the information measure.

3.2.2.2 Heuristics for Pruning and Augmenting NE List

In this step, the following pruning and augmenting heuristics are applied to the ranked NE list.

1) Distributional Similarity (DS): Two words are said to be distributionally similar if they appear in similar contexts. From the previous step, a list of words having high information score (Say, top t) is taken. In this step, t more words are taken and for each word, w, a vector of the size of the number of distinct words in the corpus is created. Each term in the vector represents the frequency with which it appears in the context (of three words) of word, w. It was observed that the NEs were clustered in some clusters and general words in other clusters. A cluster is tagged as a NE cluster if the average of the ranks of 50% of the top ranked word within the cluster is low (< t=2), and the words in that set are added as NEs. Also, if most of the words in the cluster have higher rank *i.e*. lower information content, they are removed them from the NE set. This heuristic is used for both augmenting and pruning the list.

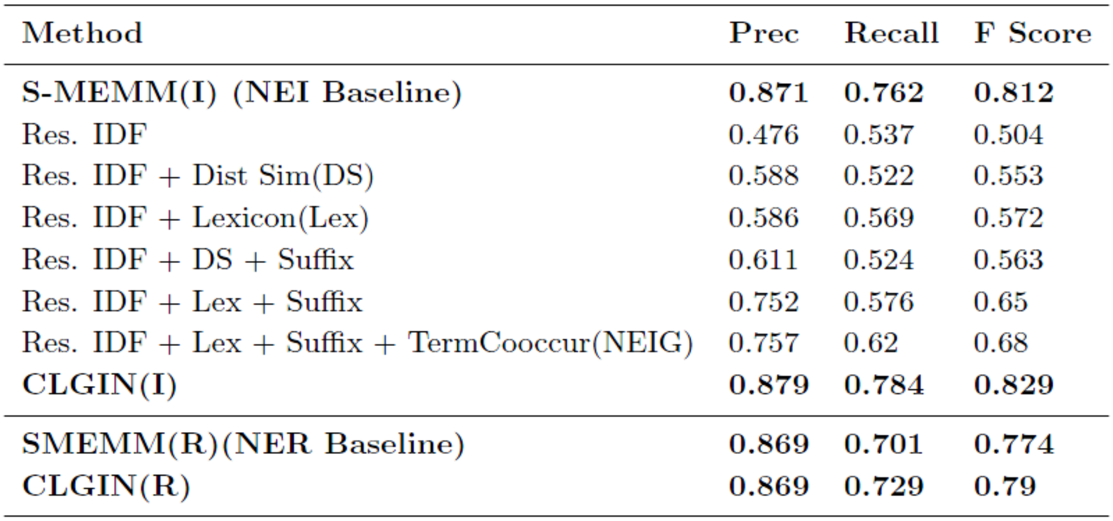
2) Lexicon: The lexicon was used as a list for excluding terms. Terms present in the lexicon have a high chance of not being NEs.

3) Suffixes: Unlike nouns, NEs usually do not take any suffixes. However, there are few exceptions like, लाल किले के बाहर (laal kile ke baahar, (outside Red Fort)) or when NEs are used as common nouns, देश को गांधियों की ज़रुरत है (desh ko gandhiyon ki zaroorat hai, The country needs Gandhis.) *etc*. Words appearing with some common suffixes like ओं (on), येंगे (yenge), *etc*. are removed from the NE list.

4) Term Co-occurrence: Co-occurrence Statistics are used to detect multiword NEs. A word may be an NE in some context but not in another. *E.g* महात्मा (mahatma “saint") when appearing with गाँधी (Gandhi \Gandhi") is a NE, but may not be, otherwise. To identify such multiword NEs, this heuristic is used. The list of NEs obtained at this step is used to tag the dataset.

3.2.3 Performance Comparison of NEIG and CLGIN Approaches (Training and Test Set from Similar Genre)

Table 4.1 compares the results of S-MEMM, NEIG and CLGIN. Besides, it also shows the stepwise improvement of NEIG approach when different heuristics were used. Identification performance of (i) Baseline System was 81.2%, (ii) NEIG was 68% and (iii) CLGIN(I) was 82.9%. Recognition performance of (i) Baseline was 77.4% and (ii) CLGIN(R) was 79%. Thus, CLGIN improved over the baseline, for both NEI and NER.



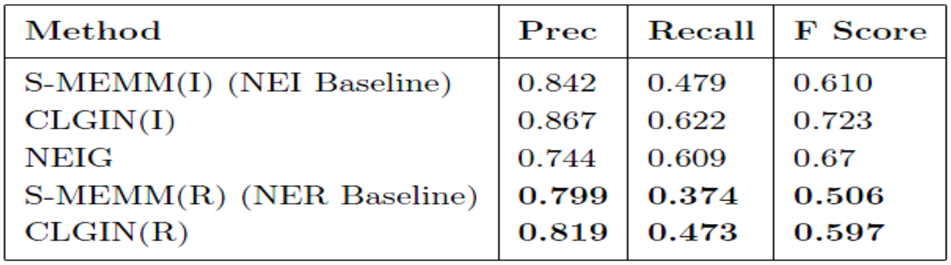
Method	Prec	Recall	F Score
S-MEMM(I) (NEI Baseline)	**0.871**	**0.762**	**0.812**
Res. IDF	0.476	0.537	0.504
Res. IDF + Dist Sim(DS)	0.588	0.522	0.553
Res. IDF + Lexicon(Lex)	0.586	0.569	0.572
Res. IDF + DS + Suffix	0.611	0.524	0.563
Res. IDF + Lex + Suffix	0.752	0.576	0.65
Res. IDF + Lex + Suffix + TermCooccur(NEIG)	0.757	0.62	0.68
CLGIN(I)	**0.879**	**0.784**	**0.829**
SMEMM(R)(NER Baseline)	**0.869**	**0.701**	**0.774**
CLGIN(R)	**0.869**	**0.729**	**0.79**

Table 3.4: Performance Comparison (similar Train and Test) (Last 2 rows are for NER; rest for NEI)

3.2.4 Performance Comparison of Baseline, NEIG and CLGIN (Training and Test Data from different genre)

Documents were randomly placed into different splits. Gyaan Nidhi is a collection of various books on several topics. Random picking resulted into the mixing of the documents, with each split containing documents from all books. But, in this experiment, we divided documents into two groups such that documents from few books (genre: Story and History) were placed into one group and rest into another group (Genre: Biography and Essay). Table 4.2 compares the

NEIG and CLGIN approaches with S-MEMM and shows that the CLGIN results are significantly better than the Baseline System.



Method	Prec	Recall	F Score
S-MEMM(I) (NEI Baseline)	0.842	0.479	0.610
CLGIN(I)	0.867	0.622	0.723
NEIG	0.744	0.609	0.67
S-MEMM(R) (NER Baseline)	**0.799**	**0.374**	**0.506**
CLGIN(R)	**0.819**	**0.473**	**0.597**

Table 3.5: Performance of various Approaches (train and test are from different genre)

The results show that adding the global information with the local context helps improve the tagging accuracy especially when the train and test data are from different genre.

3.2 Unsupervised Approach for NER : Clustering Based on Distributional Similarity

The global approaches described in earlier section aimed at Entity Identification". This section describes an approach which would be useful in tagging the identified Entities. This approach clusters together, the entities of similar types. If two words are distributionally similar, there is a high probability that these words can be replaced by each other in a sentence, without affecting the plausibility of the sentence. [Lin98] describes automatic retrieval of similar words using distributional similarity. We expect a group of related words to have similar context. In this case, the groups are expected to be a group of city names, organization names, etc.

3.2.1 The Process

This step takes the Named Entities as input and the task is to tag them with appropriate tags. The approach here, for tagging the entities is, to first cluster entities of similar types and then tag these clusters. We describe the first part in this section. The steps followed were:

1. The Yahoo! News Data was obtained and non-English documents were removed using the Wordnet. All documents which had more than 90% non-Wordnet words were tagged as non-English documents
2. The data was tagset using Stanford NER.
3. All the NE words were clubbed together and were given appropriate IDs
4. Then, the vectors were calculated for each word and each word was represented in the vector format with its TFIDF value
5. Lastly, clustering was done based on distributional cosine similarity.

3.2.2 Results

As expected, person names got clustered together. Similar was the case, with organization names and location names. When about 9000 words were clustered, around 1000 clusters were formed. The tables 3.6 and 3.7 show two sample clusters

The left side of the table are the IDs based on the tags given by the Stanford Tagger and on the right side are the actual words. There are a few words where the Stanford Parser has given wrong tags (*E.g*. NE LOCATION 3492 in Cluster 2). As can be seen from the tables, entities of similar type have got clustered together. Now these named entities need to be tagged

Word ID	Word
NE PERSON 483894	bryan Seymour
NE PERSON 234159	Schapelle
NE PERSON 213298	agathaberg langfinger
NE PERSON 59415	schapelle corby
NE PERSON 213296	ms corby
NE PERSON 213294	jodie power
NE LOCATION 4288	Bali

Table 3.6 :Cluster1

NE PERSON 152402	lopez jaen
NE LOCATION 3492	Karlsson
NE PERSON 2649	oliver wilson
NE PERSON 5003	Jimenez
NE PERSON 10466	paolo sorrentino
NE PERSON 130215	Sorrentino
NE PERSON 10467	giulio andreotti
NE PERSON 10465	matteo garrone

Table 3.7:Cluster 2

Summary

We started with introducing the Named Entity Recognition task in chapter1. We gave wide-ranging applications where NER is useful. We also explained complexities related to NER for Indian languages.

In Chapter 2, we then gave various approaches that have been developed over time for Named Entity Recognition. We highlighted the work across various languages and textual genres in NER. We described various features that are useful for rule-based as well as machine learning NER systems. The various machine learning techniques applicable to NER were also described. Various statistical measure useful for NEr were introduced. We explained how each of them worked and how they were useful in detecting anemd entities.

Finally, in Chapter 3 we gave a brief overview of the work done at IIT Bombay related to NER. We explained the working of the CLGIN system.

References

[ALF02] Alfonseca, Enrique; Manandhar, S. 2002. An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery. In *Proc. International Conference on General WordNet*.

[ASA03] Asahara, Masayuki; Matsumoto, Y. 2003. Japanese Named Entity Extraction with Redundant Morphological Analysis. In *Proc. Human Language Technology conference – North American chapter of the Association for Computational Linguistics.*

[BCK04] Bick, Eckhard 2004. A Named Entity Recognizer for Danish. In *Proc. Conference on Language Resources and Evaluation*.

[BIK97] Bikel, Daniel M.; Miller, S.; Schwartz, R.; Weischedel, R. 1997. Nymble: a High Performance Learning Name-finder. In *Proc. Conference on Applied Natural Language Processing*.

[BLK98] Black, William J.; Rinaldi, F.; Mowatt, D. 1998. Facile: Description of the NE System used for Muc-7. In *Proc. Message Understanding Conference*.

[BO74] A. Bookstein and D. R. Swanson. Probabilistic models for automatic indexing. Journal of the American Society for Information Science, 25(5):312–318, 1974.

[BOD00] Bodenreider, Olivier; Zweigenbaum, P. 2000. Identifying Proper Names in Parallel Medical Terminologies. *Stud Health Technol Inform* 77.443-447, Amsterdam: IOS Press.

[BOR98] Borthwick, Andrew; Sterling, J.; Agichtein, E.; Grishman, R. 1998. NYU: Description of the MENE Named Entity System as used in MUC-7. In *Proc. Seventh Message Understanding Conference*.

[BOU00] Boutsis, Sotiris; Demiros, I.; Giouli, V.; Liakata, M.; Papageorgiou, H.; Piperidis, S. 2000. A System for Recognition of Named Entities in Greek. In *Proc. International Conference on Natural Language Processing*

[BRN98] Brin, Sergey. 1998. Extracting Patterns and Relations from the World Wide Web. In *Proc. Conference of Extending Database Technology. Workshop on the Web and Databases*.

[BS74] A. Bookstein and D. R. Swanson. Probabilistic models for automatic index- ing. Journal of the American Society for Information Science, 25:312-318, 1974.

[CCR02] Chris Clifton, Robert Cooley, and Jason Rennie. Topcat: Data mining for topic identi_cation in a text corpus. In In Proceedings of the 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases, 2002.

[CG95] Kenneth Church and William A. Gale. Inverse document frequency (idf): A measure of deviations from poisson. In Proceedings of the Third Workshop on Very Large Corpora, pages 121{130, 1995}.

[CHE96] Chen, H. H.; Lee, J. C. 1996. Identification and Classification of Proper Nouns in Chinese Texts. In *Proc. International Conference on Computational Linguistics*.

[CHI98] Chinchor, N. (1998). MUC-7 Named Entity Task Definition Dry Run Version, Version 3.5 17 September 1997. *Proceedings of the Seventh Message Understanding Conference (MUC-7) (to appear)*. Fairfax, Virginia: Morgan Kaufmann Publishers, Inc. URL: ftp://online.muc.saic.com/NE/training/guidelines/NE.task.def.3.5.ps.

[CIM05] Cimiano, Philipp; Völker, J. 2005. Towards Large-Scale, Open-Domain and Ontology-Based Named Entity Classification. In *Proc. Conference on Recent Advances in Natural Language Processing*.

[COL99] Collins, Michael; Singer, Y. 1999. Unsupervised Models for Named Entity Classification. In *Proc. of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

[CRR03] Carreras, Xavier; Márques, L.; Padró, L. 2003. Named Entity Recognition for Catalan Using Spanish Resources. In *Proc. Conference of the European Chapter of Association for Computational Linguistic*.

[CUC01] Cucchiarelli, Alessandro; Velardi, P. 2001. Unsupervised Named Entity Recognition Using Syntactic and Semantic Contextual Evidence. *Computational Linguistics* 27:1.123-131, Cambridge: MIT Press.

[CCZ99] Cucerzan, Silviu; Yarowsky, D. 1999. Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence. In *Proc. Joint Sigdat Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

[CHU95] K. W. Church and W. A. Gale. Poisson mixtures. Journal of Natural Language Engineering, 1995.

[CLI99] C. Clifton and R. Cooley. TopCat: Data mining for topic identification in a text corpus. In Proceedings of the 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases, 1999.

[COH04] Cohen, William W.; Sarawagi, S. 2004. Exploiting Dictionaries in Named Entity Extraction: Combining Semi-Markov Extraction Processes and Data Integration Methods. In *Proc. Conference on Knowledge Discovery in Data*.

[DOD04] Doddington, George; Mitchell, A.; Przybocki, M.; Ramshaw, L.; Strassel, S.; Weischedel, R. 2004. The Automatic Content Extraction (ACE) Program – Tasks, Data, and Evaluation. In *Proc. Conference on Language Resources and Evaluation*.

[ETZ05] Etzioni, Oren; Cafarella, M.; Downey, D.; Popescu, A.-M.; Shaked, T.; Soderland, S.; Weld, D. S.; Yates, A. 2005. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *Artificial Intelligence* 165.91-134, Essex: Elsevier Science Publishers.

[FL01] Fleischman, Michael. 2001. Automated Subcategorization of Named Entities. In *Proc. Conference of the European Chapter of Association for Computational Linguistic*

[GRI96] Grishman, Ralph; Sundheim, B. 1996. Message Understanding Conference - 6: A Brief History. In *Proc. International Conference on Computational Linguistics*.

[HA75] S. P. Harter. A probabilistic approach to automatic keyword indexing: Part I. On the distribution of specialty words in a technical literature. Journal of the American Society for Information Science, 26(4):197–206, 1975.

[HNG05] Huang, Fei. 2005. *Multilingual Named Entity Extraction and Translation from Text and Speech*. Ph.D. Thesis. Pittsburgh: Carnegie Mellon University.

[HEN06] Heng, Ji; Grishman, R. 2006. Data Selection in Semi-supervised Learning for Name Tagging. In *Proc. joint conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics. Information Extraction beyond the Document.*

[HOV02] Fleischman, Michael; Hovy. E. 2002. Fine Grained Classification of Named Entities. In *Proc.Conference on Computational Linguistics*.

[Jon72] Karen Sprck Jones. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, 28:11-21, 1972.

[JON73] K. S. Jones. Index term weighting. Information Storage and Retrieval, 9:619–633, 1973.

[JWN03] Jwnl. java wordnet library jwnl 1.3. http://sourceforge.net/projects/jwordnet/, 2003.

[KKK98] Kokkinakis, Dimitri. 1998., AVENTINUS, GATE and Swedish Lingware. In *Proc. of Nordic Computational Linguistics Conference*.

[LEE05] Lee, Seungwoo; Geunbae Lee, G. 2005. Heuristic Methods for Reducing Errors of Geographic Named Entities Learned by Bootstrapping. In *Proc. International Joint Conference on Natural Language Processing*.

[LI07] : David Nadeau, Satoshi Sekine .A survey of named entity recognition and classification : Lingvisticae Investigationes, Vol. 30, No. 1. (January 2007), pp. 3-26,

[Lin98] Dekang Lin. Automatic retrieval and clustering of similar words. In Pro- ceedings of the 17th international conference on Computational linguistics, pages 768{774, Morristown, NJ, USA, 1998. Association for Computational Linguistics.

[LM03] Wei Li and Andrew McCallum. Rapid development of hindi named entity recognition using conditional random _elds and feature induction. ACM Transactions on Asian Language Information Processing (TALIP), 2(3):290-294, 2003.

[MAY03] May, Jonathan; Brunstein, A.; Natarajan, P.; Weischedel, R. M. 2003. Surprise! What's in a Cebuano or Hindi Name? *ACM Transactions on Asian Language Information Processing* 2:3.169-180, New York: ACM Press

[MCM03] McCallum, Andrew; Li, W. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Features Induction and Web-Enhanced Lexicons. In *Proc. Conference on Computational Natural Language Learning*.

[MIN05] Minkov, Einat; Wang, R.; Cohen, W. 2005. Extracting Personal Names from Email: Applying Named Entity Recognition to Informal Text. In *Proc. Human Language Technology and Conference Conference on Empirical Methods in Natural Language Processing*.

[MKV99] Mikheev, A.; Moens, M.; Grover, C. 1999. Named Entity Recognition without Gazetteers. In *Proc. Conference of European Chapter of the Association for Computational Linguistics*.

[MYD01] Maynard, Diana; Tablan, V.; Ursu, C.; Cunningham, H.; Wilks, Y. 2001. Named Entity Recognition from Diverse Text Types. In *Proc. Recent Advances in Natural Language Processing*.

[NAD06] Nadeau, David; Turney, P.; Matwin, S. 2006. Unsupervised Named Entity Recognition: Generating Gazetteers and Resolving Ambiguity. In *Proc. Canadian Conference on Artificial Intelligence*.

[Nal04] Ramesh Nallapati. Discriminative models for information retrieval, 2004.

[NAR03] Narayanaswamy, Meenakshi; Ravikumar K. E.; Vijay-Shanker K. 2003. A Biological Named Entity Recognizer. In *Proc. Pacific Symposium on Biocomputing*.

[OHT02] Ohta, Tomoko; Tateisi, Y.; Kim, J.; Mima, H.; Tsujii, J. 2002. The GENIA Corpus: An Annotated Research Abstract Corpus in Molecular Biology Domain. In *Proc. Human Language Technology Conference*.

[PAS06] Pasca, Marius; Lin, D.; Bigham, J.; Lifchits, A.; Jain, A. 2006. Organizing and Searching the World Wide Web of Facts—Step One: The One-Million Fact Extraction Challenge. In *Proc. National Conference on Artificial Intelligence*.

[PAP01] Kishore Papineni. Why inverse document frequency? In NAACL '01: Second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies 2001, pages 1{8, Morristown, NJ, USA, 2001. Association for Computational Linguistics.

[PET01] Petasis, Georgios; Vichot, F.; Wolinski, F.; Paliouras, G.; Karkaletsis, V.; Spyropoulos, C. D. 2001. Using Machine Learning to Maintain Rule-based Named-Entity Recognition and Classification Systems. In *Proc. Conference of Association for Computational Linguistics*.

[PIS04] Piskorski, Jakub. 2004. Extraction of Polish Named-Entities. In *Proc. Conference on Language Resources an Evaluation*.

[PLM 97] Palmer, David D.; Day, D. S. 1997. A Statistical Profile of the Named Entity Task. In *Proc. ACL Conference for Applied Natural Language Processing*.

[POI01] Poibeau, Thierry; Kosseim, L. 2001. Proper Name Extraction from Non-Journalistic Texts. In *Proc. Computational Linguistics in the Netherlands*.

[POI03] Poibeau, Thierry. 2003. The Multilingual Named Entity Recognition Framework. In *Proc. Conference on European chapter of the Association for Computational Linguistics.*

[POP04] Popov, Borislav; Kirilov, A.; Maynard, D.; Manov, D. 2004. Creation of reusable components and language resources for Named Entity Recognition in Russian. In *Proc. Conference on Language Resources and Evaluation*.

[RAG04] Raghavan, Hema; Allan, J. 2004. Using Soundex Codes for Indexing Names in ASR documents. In *Proc. Human Language Technology conference - North American chapter of the Association for Computational Linguistics. Interdisciplinary Approaches to Speech Indexing and Retrieval*.

[Rau91] L. F. Rau. Extracting company names from text. In Artificial Intelligence Applications, 1991. Proceedings., Seventh IEEE Conference on, volume i, pages 29:32, 1991.

[RIL99] Riloff, Ellen; Jones, R 1999. Learning Dictionaries for Information Extraction using Multi-level Bootstrapping. In *Proc. National Conference on Artificial Intelligence*.

[RIN05] Rindfleisch, Thomas C.; Tanabe, L.; Weinstein, J. N. 2000. EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature. In *Proc. Pacific Symposium on Biocomputing*.

[RJ05] Jason D. M. Rennie and Tommi Jaakkola. Using term informativeness for named entity detection. In SIGIR '05: Proceedings of the 28th annual inter- national ACM SIGIR conference on Research and development in informa- tion retrieval, pages 353{360, New York, NY, USA, 2005. ACM.

[SAN06] Santos, Diana; Seco, N.; Cardoso, N.; Vilela, R. 2006. HAREM: An Advanced NER Evaluation Contest for Portuguese. In *Proc. International Conference on Language Resources and Evaluation*.

[SEK98] Sekine, Satoshi. 1998. Nyu: Description of the Japanese NE System Used For Met-2. In *Proc. Message Understanding Conference*.

[SEK00] Sekine, Satoshi; Isahara, H. 2000. IREX: IR and IE Evaluation project in Japanese. In *Proc. Conference on Language Resources and Evaluation*.

[SEK04] Sekine, Satoshi; Nobata, C. 2004. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. In *Proc. Conference on Language Resources and Evaluation*.

[SH10] Shalini Gupta, Pushpak Bhattacharyya Think globally, apply locally: using distributional characteristics for Hindi named entity identification NEWS '10 Proceedings of the 2010 Named Entities Workshop

[SLV04] Da Silva, Joaquim Ferreira; Kozareva, Z.; Lopes, G. P. 2004. Cluster Analysis and Classification of Named Entities. In *Proc. Conference on Language Resources and Evaluation*.

[SSM08] Sujan Kumar Saha, Sudeshna Sarkar, and Pabitra Mitra. A hybrid feature set based maximum entropy hindi named entity recognition. In Proceedings of the Third International Joint Conference on Natural Language Processing, Kharagpur, India, 2008.

[TJ02] Tjong Kim Sang, Erik. F. 2002. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *Proc. Conference on Natural Language Learning.*

[TJ03] Tjong Kim Sang, Erik. F.; De Meulder, F. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proc. Conference on Natural Language Learning*.

[WAN92] Wang, Liang-Jyh; Li, W.-C.; Chang, C.-H. 1992. Recognizing Unregistered Names for Mandarin Word Identification. In *Proc. International Conference on Computational Linguistics*.

[WHI03] Whitelaw, Casey; Patrick, J. 2003. Evaluating Corpora for Named Entity Recognition Using Character-Level Features. In *Proc. Australian Conference on Artificial Intelligence*.

[WOL95] Wolinski, Francis; Vichot, F.; Dillet, B. 1995. Automatic Processing Proper Names in Texts. In *Proc. Conference on European Chapter of the Association for Computational Linguistics*.

[WTT99] Witten, Ian. H.; Bray, Z.; Mahoui, M.; Teahan W. J. 1999. Using Language Models for Generic Entity Extraction. In *Proc. International Conference on Machine Learning. Text Mining*.

[YU98] Yu, Shihong; Bai S.; Wu, P. 1998. Description of the Kent Ridge Digital Labs System Used for MUC-7. In *Proc. Message Understanding Conference*.

[ZHU05] Zhu, Jianhan; Uren, V.; Motta, E. 2005. ESpotter: Adaptive Named Entity Recognition for Web