Information Extraction for CLIA System

Kavitha. V, Vijayakrishna. R and Sobha, Lalitha Devi.
AU-KBC Research Centre, Anna University, Chennai
{Kavitha, sobha}@au-kbc.org

Information Extraction (IE) is the automatic extraction of structured information such as entities and relationships between entities from unstructured sources. The extraction of information from noisy, unstructured sources is a challenging task. In extracting the entity and the relationship between entities we use two approaches a) Entity extraction and b) Relation extraction. While IE helps in enhancing the ranking in monolingual information retrieval it helps in ranking as well as in transferring the information into other languages without a full translation system in cross lingual information retrieval. In this paper, we give in detail the IE approach we have adopted in 'India Search' CLIA system.

Introduction

Information Extraction is a process that takes unstructured texts as input and produces fixed-formatted unambiguous data as output. It automatically extracts the structured information such as entities and relationships between entities from unstructured texts. This helps in using better forms of queries for retrieving relevant documents from a pool of unstructured texts than with keyword searches. The extraction of information from noisy, unstructured test is a challenging task.

Extracting information from text as a demonstration of "understanding" goes back to the early days of Natural Language Understanding (NLP). Early extraction tasks were concentrated around the identification of named entities, like people and company names and relationship from text. The two competitions, the Message Understanding Conference (MUC) and Automatic Content Extraction (ACE) program increased the scope of this task. Early systems were rule-based with manually coded rules [3]. Since manual coding of rules are tedious, algorithms for automatically learning rules from examples were developed [2, 22]. As extraction systems were targeted on more noisy unstructured sources, rules were found to be weak. With the advent of statistical learning, two parallel kinds of techniques were deployed: generative models based on Hidden Markov Models [1, 7, 21] and conditional models based on maximum entropy [17,18, 19]. Much of the work in IE has confined itself to certain domains though attempts to build generic IE systems are also researched upon. Though there are many techniques, there is no one technique which could be said as the best approach for this task. Rule-based methods and statistical methods are used in parallel depending on the nature of the extraction task. There also exist hybrid models that are used to get the benefits of both statistical and rule-based methods. Thus Information extraction has made significant progress in the last decade and three broad strategies have emerged towards this end: 1) the stochastic 2) Rule Based 3) Knowledge based approach.

The information needs to be extracted can be of different type such as entities, relationships between entities, attributes describing entities etc. There are different approaches for extracting

each of the above type of information. The entities are extracted using Named Entity recognizer and the relationships and attribute describing entities are extracted using syntactic rules. In this paper we detail the approaches used in the CLIA system for information extraction from English tourism documents viz A) Named entity extraction and B) entity relation extraction.

A. Named Entity Recognition

Named Entity Recognition (NER) is the task of identifying and classifying the entities such as person names, place names, organization names etc, in a given document. Named entities play a major role in information extraction. NER has been a defined subtask in Message Understanding Conference (MUC) since MUC 6. A well performing NER is important for further level of NLP techniques.

Many techniques have been applied in Indian and European languages for NER. Some of them are rule based system [12], which makes use of dictionary and patterns of named entities, Decision trees [11], Hidden Morkov Model (HMM) [5], Maximum Entropy Morkov Model (MEMM) [6], Conditional Random Fields (CRF) etc. In short, the approaches can be classified as rule-based approach, machine learning approach or hybrid approach. For Indian languages, many techniques have been experimented by different people. In that the MEMM system for Hindi NER [13] gave an average F1 measure of 71.9 for a tagset of four named entity tags.

NER has been done generically and also domain specific where a finer tagset is needed to describe the named entities in a domain. Domain specific NER is common and has been in existence for a long time in the Bio-domain [20] for identification of protein names, gene names, DNA names etc.

We have used Conditional Random Fields, a machine learning approach to sequence labeling task, which includes NER. We have also developed a domain specific hierarchical tagset consisting of 106 tags for tourism and health domain.

2.1 Conditional Random Fields (CRF)

Conditional Random Fields (CRF) [15] is a machine learning technique. CRF overcomes the difficulties faced in other machine learning techniques like Hidden Markov Model (HMM) [16] and Maximum Entropy Markov Model (MEMM) [4]. HMM does not allow the words in the input sentence to show dependency among each other. MEMM shows a label bias problem because of its stochastic state transition nature. CRF overcomes these problems and performs better than the other two. HMM, MEMM and CRF are suited for sequence labeling task. But only MEMM and CRF allows linguistic rules or conditions to be incorporated into machine learning algorithm.

[15] define Conditional Random Fieds as follows**:** "Let $\mathbf{G} = (\mathbf{V},\mathbf{E})$ be a graph such that $\mathbf{Y} = (\mathbf{Y}_v)_{v\hat{\imath}V}$, so that $\mathbf{Y}$ is indexed by the vertices of $\mathbf{G}$. Then $(\mathbf{X},\mathbf{Y})$ is a conditional random field in case, when conditioned on $\mathbf{X}$, the random variables $\mathbf{Y}_v$ obey the Markov property with respect to the graph: $p(\mathbf{Y}_v|\mathbf{X},\mathbf{Y}_w,w\neq v) = p(\mathbf{Y}_v|\mathbf{X},\mathbf{Y}_w,w\sim v)$, where w~v means that w and v are neighbors in G".

Here $\mathbf{X}$ denotes a sentence and $\mathbf{Y}$ denotes the label sequence. The label sequence $\mathbf{y}$ which maximizes the likelihood probability $p_{\Theta}(\mathbf{y}|\mathbf{x})$ will be considered as the correct sequence, while testing for new sentence $\mathbf{x}$ with CRF model Θ. The likelihood probability $p_{\Theta}(\mathbf{y}|\mathbf{x})$ is expressed as follows.

$$p_\theta(\mathrm{y} \mid \mathrm{x}) \propto \exp\left(\sum_{e \in E,k} \lambda_k \, f_k(e, \mathrm{y}|_e, \mathrm{x}) + \sum_{v \in V,k} \mu_k \, g_k(v, \mathrm{y}|_v, \mathrm{x})\right)$$

where λ_k and μ_k are parameters from CRF model θ and f_k and g_k are the binary feature functions that we need to give for training the CRF model. This is how we integrate linguistic features into machine learning models like CRF.

2.3 Named Entity Tagset

The tagset which we use here for NER contains 106 tags related to each other hierarchically. This type of tagset is motivated from "ACE English Annotation Guidelines for Entities" developed by Linguistic Data Consortium. The tagset which we use is built in-house with focus to tourism and health domain.

2.4 Sample Tags

Sample tags from the entire tagset is shown below with their hierarchy.

1. Enamex
   - 1.1. Person
     - 1.1.1. Individual
       - 1.1.1.1. Family Name
       - 1.1.1.2. Title
     - 1.1.2. Group
   - 1.2. Organization
2. Numex
3. Timex

Certain tags in this tagset are designed with focus to tourism and health domain, such as place, address, water bodies (rivers, lakes etc.,), religious places, museums, parks, monuments, airport, railway station, bus station, events, treatments for diseases, distance and date.

The tags are assigned with numbers 1,2,3 for zeroth level, the tags with numbers 1.1, 1.11, 2.1 ,2.4 and 3.1 ,3.7 for level-1 , the tags with numbers 1.1.1, 1.1.2, 1.2.1 etc as level-2 and the tags with numbers 1.1.1.1, 1.1.1.2, 1.2.4.1 etc for level-3 because they occur in the hierarchy in corresponding levels. We have 25 tags in level-1, 50 in level-2 and 31 in level-3.

2.4 Sample Annotation

<person> <city> Madhurai </city> <individual> Mani <familyname> Iyer </ familyname> </individual> </person> came to <city> Chennai </city>.

2.5 NER using CRF

We used CRF++ [23], an open source toolkit for linear chain CRF. This tool when presented with the attributes extracted from the training data builds a CRF model with the feature template specified by us. When presented with the model thus obtained and attributes extracted from the test data, CRF tool outputs the test data tagged with the labels that has been learnt.

2.6 Presenting training data

Training data will contain nested tagging of named entities. To handle nested tagging and to avoid ambiguities, we isolate the tagset into three subsets, each of which will contain tags from one level in the hierarchy. Now, the training data itself will be presented to CRF as three sets of training data. From this, we will get three CRF models, one for each level of hierarchy.

Example:

The sample sentence given in 2.4 will be presented to CRF training for each level of hierarchy as follows:

Level-1:

<location> Madhurai <location> <personl> Mani Iyer </person> came to <location> Chennai </location>.

Level-2:

<place> Madhurai </place> <individual> Mani > Iyer </individual> came to <place> Chennai </place>.

Level-3:

<city> Madhurai </city> Mani <familyname> Iyer </familyname> came to <city> Chennai </city>.

Notice that the tags 'location' and 'place' are not specified in the input sentence. In the hierarchy, the 'location' tag is the parent tag of 'place' tag which is a parent tag of 'city' tag. Thus for the word "Madhurai", level-1 tag is 'location', level-2 tag is 'place' and level-3 tag is 'city'.

2.6 Attributes and Feature Templates

Attributes are the dependencies from which the system can infer a phrase to be named entity or not. Features are the conditions imposed on these attributes. Feature templates help CRF engine to form features from the attributes of the training data. From the characteristics of named entities in Tamil, we see that it is only the noun phrases that are possible candidates for Named Entities. So we apply Noun Phrase Chunking and consider only noun phrases and train on them. The attributes that we arrived at are explained below:

1. Root of word: This is to ignore inflections in named entities. Here we take a window of 5 words and consider unigram, bigram and trigram combinations. This will capture the context in which a named entity will occur.
2. Parts of Speech (POS): This will give whether a noun is proper noun or common noun. POS of current word is considered.
3. Words and POS combined: Present word combined with POS of previous two words and present word combined with POS of the next two words are taken as features.
4. Dictionary of Named Entities: A list of named entities is collected for each type of named entities. Root words are checked against the dictionary and if present in the dictionary, the dictionary feature for the corresponding type of named entity is considered positive.
5. Patterns: Certain types of named entities such as date, time, money etc., show patterns in their occurrences. These patterns are listed out. The current noun phrase is checked against each pattern. The feature is taken as true for those patterns which are satisfied by the current noun phrase.
6. Bigram of Named Entity label: A feature considering the bigram occurrences of the named entity labels in the corpus is considered. This is the feature that binds the consecutive named entity labels of a sequence and thus forming linear chain CRFs.

2.7 Presenting testing data

Test data is processed for POS and NP chunking. Here also, the same set of attributes and feature templates are used. Now, the test data is tagged with each of the CRF models built for three levels of hierarchy. All the three outputs are merged to get a combined output.

3 Experiments

One lakh word corpus is collected in tourism domain. POS tagging, NP chunking and named entity annotation are done manually on the corpus. This corpus contains about 23k named entities. The corpus is split into two sets. One forms the training data and the other forms the test data. They consist of 80% and 20% of the total data respectively. CRF is trained with training data and CRF models for each of the

levels in the hierarchy are obtained. With these models the test data is tagged and the output is evaluated manually.

3.1 Results

The results of the above experiment are as follows. Here, NE means Named Entity, NP means noun phrase. Number of NPs in test data = 7922

There are totally 4059 NEs in the test data. All of them bear level-1 tags. Out of 4059 NEs, 3237 NEs bear level-2 tags and 727 NEs bear level-3 tags. The result from the system is shown in Table-1.

Named Entity Level	Level-1	Level-2	Level-3
Number of NEs in data	4059	3237	727
Number of NEs identified by NER engine	3414	2667	606
Number of NEs identified correctly	3056	2473	505
Precision %	89.51	**92.73**	83.33
Recall %	75.29	**76.40**	69.46
F1 measure %	81.79	**83.77**	75.77

Overall result

Performance Measure	Value in %
Precision	88.52
Recall	73.71
F1 Measure	80.44

Table-1: Evaluation of output from NER engine

The system performs well for domain focused corpus. The reason for good precision is that tagging is done only when the root word that is seen is already learnt from the training corpus or the context of the current word is similar to the context of the named entities that it has learnt from the training corpus.

When there are new named entities which are not in training corpus, the system tries to capture the context and tags accordingly. In such cases irrelevant context that it may learn while training will cause problem resulting in wrong tagging. This affects the precision to some extent. When the named entities and their context are new, then they are most likely not tagged. This affects the recall.

3 Entity Relation Extraction

We use a rule based method for extracting the relation information of entities. The templates used for extraction are pre-defined and we have defined 11 templates for Tourism domain which covers most of the information required for a tourist. Each template contains template elements and the relational elements. The template elements are derived from the NE tags and the relational elements are associated with the domain specific verb. The verbs give the relation between the entities and thus verb play a major role in relational extraction. The rules for extraction are verb centric.

B. Relation Extraction

In IE, the extraction centres on four Wh-questions “Who, What, Where and When”. How exact answers for these questions can be found will depend on the approach used for extraction. Here we are using a rule based approach for extracting information from English documents from tourism domain.

4.1 Relational extraction Rules

The rules are of two types, generic rules

which can be used across domains and specific rules which are domain specific. The specific rules depend on the verbs and they are crafted according to the verb and its arguments. Other these rules heuristic rules are also used both domain independent and specific. A domain specific verb dictionary is required for identifying the position of the information to be extracted and to decide what information to be extracted. This dictionary is made from the corpus. In this work we have only one generic rule and more specific rules. The generic rule we used is any verb and its argument

R1. [NP] {verb}[NP]

For each specific rule we consider each verb and what could be its arguments are identified. Since we are not using indepth parsing of the text, the rules depend on other grammatical features such as adjectives, pre-position etc and named entity details. Consider the following rules

Verbs: famous, popular and constructed.

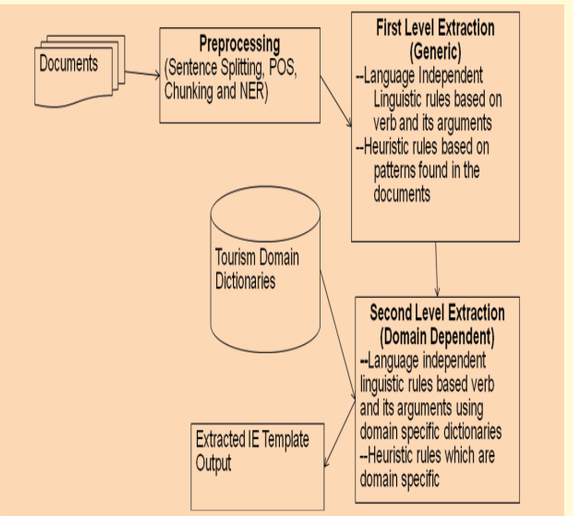
R2. [NP] {in [NP]} is [famous/popular] for [NP]
<place> <specialty>
Ex : Baguru in Rajastan is famous for its block
paniting.

R3. [NP] was constructed {by [NP]} in [NP]
<location> <period>
Ex : Taj Mahal was constructed by shahjahan, in
mughal's period in remembrance of his favourite wife ...

Verbs: Located and Situated

R3. [located/situated] at an [altitude/elevation/height/longitude] of <num> feet/m [from/above/below] [sea level/surface]
Ex : Ghoom Senchal Ridge situated at an average altitude of 7000ft above the sea,

Similar to the above rules, we have crafted 75 rules for populating 11 templates (details given in Annexure I). The input to the extraction system is the text pre-processed for POS, NP Chunking, Clause identified and NE identified text. The rules are applied to extract information from these pre-processed texts. The system Architecture is given below.



4.2 Evaluation

We have taken three sets of documents from the Tourism corpus and evaluated the system. The below table gives the results

No. of Documents	No. of extraction templates possible	No. of templates Extracted	Correctly Identified Templates	Accuracy (%)
Set 1- 100 docs	**200**	**170**	**130**	**76.4**
Set 2 - 200 docs	**500**	**456**	**380**	**83.63**
Set 3 - 200 docs	**650**	**580**	**490**	**84.4**
Average				**81.36**

Table 2: IE evaluation results of the IE system.

Though the rules are covering all types of patterns available in the testing corpus the error

is occurring mainly due to the inaccuracy of the preprocessing modules. In most of the cases the chunker gave wrong tag and it has affected in extracting the correct noun phrase. Though we have handled the passive construction of the verb, the rules handling this construction require fine tuning. Another main problem we encountered is in the input. The input data had unwanted text. The cleaning of the data is required for improving the accuracy.

5. Conclusion.

The Information Extraction module working in CLIA system is given in detail in this paper. The paper elaborated two modules of extraction NER and Relation Extraction. The NER are performs with 80.44% and the Relational Extraction performs with 81.36%.

Acknowledgment

The work has been carried out as part of the Department of Electronics & Information Technology (Deity), Government of India funded Consortium Project "Development of Cross Lingual Information Access (CLIA)" System.

References

1. E. Agichtein and V. Ganti, "Mining reference tables for automatic text segmentation," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, USA,
2. J. Aitken, "Learning information extraction rules: An inductive logic programming approach," in *Proceedings of the 15th European Conference on Artificial Intelligence*, pp. 355–359, 2002.
3. D. E. Appelt, J. R. Hobbs, J. Bear, D. J. Israel, and M. Tyson, "Fastus: A finite-state processor for information extraction from real-world text," in *IJCAI*, pp. 1172–1178, 1993.
4. A. Berger, S. Della Pietra and V. Della Pietra, A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, vol.22, no. 1. 1996.
5. D.M. Bikel., Nymble: a high-performance learning name-finder. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*. Pages 194-201, 1997.
6. A Borthwick, et al., Description of the MENE named Entity System, In *Proceedings of the Seventh Machine Understanding Conference* (MUC-7). 1998.
7. V. R. Borkar, K. Deshmukh, and S. Sarawagi, "Automatic text segmentation for extracting structured records," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, Santa Barabara, USA, 2001.
8. Fuchun Peng and Andrew McCallum. Accurate Information Extraction from Research Papers using Conditional Random Fields. *In the Proceedings of NAACL 2004*.
9. D.Freitag and A.McCallum. Information extraction with HMMs and shrinkage. *In AAAI 99 Workshop on Machine Learning for Information Extraction*. pp.31-36, 1999.
10. Hai Leong Chieu and Hwee Tou Ng. A Maximum Entropy Approach to Information Extraction from Semi-Structured and Free Text. *In the Proceedings of AAAI 2002*.
11. V. Karkaletsis G. Pailouras and C.D. Spyropoulos. Learning decision trees for named-entity recognition and classification. In *Proceedings of the ECAI Workshop on Machine Learning for Information Extraction, 2000*

12. G.R. Krupka and K. Hausman. Iso Quest Inc: Descriptino of the NetOwl Text Extraction System as used for MUC-7. In *Proceedings of Seventh Machine Understanding Conference* (MUC 7), 1998.

13. Kumar N and Pushpak Bhattacharya. Named Entity Recognition in Hindi using MEMM, 2006.

14. Kun Yu, Gang Guan, and Ming Zhou. Resume Information Extraction with Cascaded Hybrid Model. *In ACL 2005*

15. John Lafferty, Andrew McCallum, Fernando Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning* (ICML-2001). pages 282-289, 2001.

16. Lawrence R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Proceedings of the IEEE, 77 (2).* p. 257–286, February 1989.

17. R. Malouf, "Markov models for language-independent named entity recognition," in *Proceedings of the Sixth Conference on Natural Language Learning (CoNLL-2002)*, 2002.

18. A. McCallum, D. Freitag, and F. Pereira, "Maximum entropy markov models for information extraction and segmentation," in *Proceedings of the International Conference on Machine Learning (ICML-2000)*, pp. 591–598, Palo Alto, CA, 2000.

19. A. Ratnaparkhi, "Learning to parse natural language with maximum entropy models," *Machine Learning*, vol. 34, 1999.

20. B. Settles., Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications* (NLPBA), Geneva, Switzerland. pages 104-107, 2004.

21. K. Seymore, A. McCallum, and R. Rosenfeld, "Learning Hidden Markov Model structure for information extraction," in *Papers from the AAAI-99 Workshop on Machine Learning for Information Extraction*, pp. 37–42, 1999.

22. S. Soderland, "Learning information extraction rules for semi-structured and free text," *Machine Learning*, vol. 34, 1999.

23. Taku Kudo, CRF++, an open source toolkit for CRF,

http://crfpp.sourceforge.net.