Complex Relations Extraction

Naveed Afzal
University of Wolverhampton, UK
n.afzal@wlv.ac.uk

Abstract

Information Extraction (IE) is the process of deriving useful information from unstructured text documents. Much of the recent research in the area of IE has been focused on named identification and binary relations extraction. In this paper, we investigate the problem of complex relations extraction. Complex relations are n-ary (n>2) relations between n entities.

In this paper, we present an approach in which complex relations are first factorised into binary relations and then different classifiers (Maximum Entropy, Naïve Bayes and Decision Tree) are trained for relatedness in this paper. A graph is then created between the pairs of related entities in order to reconstruct complex relations. In the second stage, maximal cliques are used for finding potential complex relation instances in that graph.

The system is evaluated against the results obtained from different classifiers in terms of Precision, Recall and F-score. The system achieves better results when compared to other presented approaches.

1. Introduction

There is a substantial amount of data which is present in the form of natural language. This data is predominantly in an unstructured form and for the purpose of automatic manipulation and to analyse that data, there is a requirement to structure the data in a manner that makes information more comprehensible.

"Information Extraction (IE) is the identification of instances of a particular class of events or relationship in a natural language text, and the extraction of the relevant arguments of the event or relationship. Information extraction therefore involves the creation of a structured representation (such as a database) of selected information drawn from the text". [3]

Information extraction (IE) is not the equivalent of Information Retrieval (IR). In IR the documents relate to a specific query, and are searched and returned. Google, the search engine, is one of example of an IR engine. On the other hand IE have the capability to extract all the important data present in a document with high precision and accuracy.

For example

"James Anderson was appointed vice president of the Proctor & Gamble Company of London".

In the afore-mentioned example the entities we are interested in extracting are underlined and these are:

Person = James Anderson
Company = Proctor & Gamble
Post = Vice President.

Generally, a template is used to define the items of interest in a specific text.

Until now much of the research being carried out in text information extraction (IE) has mostly been dedicated to accurate tagging of the named entities. The subsequent step in IE is the extraction of relations involving the named entities. Largely the relations extraction system focuses on the particular problem of extracting binary relations. An extremely modest amount of work has been done so far in recognising and extracting more complex relations [6], [2].

Events are often used to describe connection between multiple entities. The simple relations (binary) can be explained by the following example:

"Bill Gates resigned as Microsoft's CEO."

In the above example entities of interest are underlined and a standard algorithm will easily extract the subsequent binary relations:
Person Out (Bill Gates) --- Post (CEO)
Person Out (Bill Gates) --- Org (Microsoft)

Post (CEO) ----Org (Microsoft)

The complex relations can be explained by the following example:

"Ferguson and Wenger are managers at Manchester United and Arsenal football clubs respectively."

This sentence contains two people, one job title and two companies and it is difficult to identify which person belongs to which company.

Complex relations are n-ary relations among n types entities (n>2). The aim of complex relations extraction is to identify all the instances of interest in some piece of text, including incomplete sentences.

2. Previous Work

The approach presented by [6] begins by extracting all complex relations into a set of binary relations. The general processes in this paper are that complex relations are first factored into binary relations, binary classifiers are then trained for relatedness, graph is built among related entities and complex relations are rebuild from that graph.

Another approach presented by [2] based on classification towards a single-slot (Seminar Announcement) as well as multi-slot (Management Succession) information extraction. Both IE systems were built using maximum entropy classifier.

The multi-slot IE system uses MUC6 (which is based on management succession) on a sentence-by-sentence basis, extracting templates of 4 slots. These 4 slots are person-in (a person coming into a certain position in an organisation), person-out (person leaving a certain position in an organisation), the organisation position itself and the organisation name. In order to build templates, this system requires certain domain knowledge. A text categorisation module is applied to identify the documents that do not contain any relevant templates. The documents containing relevant templates are treated as positive examples while those that do not are treated as negative examples. All the position and organisation names are tagged in candidate selection phase and a classifier is being built for each slot. The classifier is then trained using sentences from the relevant documents. Pair-wise relations among the entities are found in the relation classification phase. In the template building phase, only the templates that consist of at least a person-in or person-out are treated as valid templates. Therefore, in order to represent relations in a form of a graph an edge will exit between two entities in a sentence only and only if their relation is classified as positive. The system will then find the largest clique. The decision among cliques of same size will be taken using the highest product of probabilities of relations.

Once a template is finished, the entities that constitute that template will be isolated from the graph and the system begins with the new graph of remaining entities.

3. Approach

This paper extends the work carried out by [6]. Moreover the [6] used the biomedical text, while this paper is using an adapted version of MUC-6 data presented by [7]. So from now onward whenever MUC-6 data is mentioned in this paper it means the adapted version of MUC-6 data presented by [7].

MUC-6 [4] is related to management succession tasks in which these tasks are spread across many sentences. While an adapted version of MUC-6 data that this paper is using looks at this management succession tasks on a single sentence level. The entities of interest in MUC-6 data are the person, post and company and the relations between them. Figure 1 show the feature set used to build the feature vectors.

So in the first step, the MUC-6 data is transformed according to the feature set given in figure 1. A Feature contains the information about the class of the object we want to classify. After that each MUC-6 event is broken down into binary relations.

Feature Set
entity type of *e1* and *e2*
Words in *e1* and *e2*
Words bigrams in *e1* and *e2*
POS of *e1* and *e2*
Words between *e1* and *e2*
Words bigrams between *e1* and *e2*
POS between *e1* and *e2*
Distance between *e1* and *e2*

Figure 1: Feature set for binary relations classifier e1 and e2 are entities.

MALLET[1] is an integrated collection of Java code useful for Statistical Natural Language Processing, document classification clustering, information extraction and other machine learning applications to text. This paper is using classifiers present in MALLET for the classification of MUC-6 data. One reason for selecting the MALLET is for the highly efficient classifiers training.

A typical usage of MALLET for the classification process consists of two steps: Reading the documents into MALLET and converting them into a list of instances, where each instance is a feature vector and classifying the feature vectors. Different classifiers (Maximum Entropy, Naïve Bayes and Decision Tree) are then trained to do relations classification on MUC-6 training data and after the training they are tested on the MUC-6 testing data.

4. Experiments

The MUC-6 data is first pre-processed and then parts of speech tagging take place and at the end labelling of relations is done based on the information present in the MUC-6 data. In order to get the parts-of-speech of entities (Person, Post and Company) and the words in between them, the QTag[2] parts-of-speech tagger is used.

QTag is a probabilistic parts-of-speech tagger. It reads the text from the documents and for each token a part-of-speech (e.g. noun, verb, punctuation, etc.) is returned. It is quite robust and tags text with good accuracy because it uses the probabilities to address the ambiguity problem.

The relation between two entities is labelled as Positive or Negative depending upon the information present between *@@TAGS Succession* and *@@ENDTAGS* in the MUC-6 data. If the two entities are mention between these two tags then relation between them is labelled as Positive, otherwise it is labelled as Negative.

After the creation of list of feature vectors now we can use any classifier present in the MALLET to do the classification. MUC-6 training data is used in order to train the classifier while MUC-6 testing data is used to test the classifier. *Classification* is the process of assigning objects from a universe to two or more classes. A classifier then collects the statistics from the training set and derives internal parameters from those statistics; those parameters are later applied to classifying the validation set and output the classifications. The default output of the classifier includes accuracies, standard deviation and standard error for the training and test data, and a confusion matrix for the test data.

5. Evaluation

MUC-6 Training data consists of 8361 binary relations while testing data contains 1819 binary relations. Table 1 shows the relation classification results obtained by using Maximum Entropy, Naïve Bayes and Decision Tree classifiers.

The "Training Data Accuracy" column in Table 1 shows how effectively each classifier is trained on the training data. Maximum Entropy classifier achieves higher training data accuracy compare to both Naïve Bayes and Decision Tree classifier. While in terms of "*Testing Data Accuracy*" Decision Tree classifier performs slightly better than Maximum Entropy classifier. The results produced by the Naïve Bayes classifier are very poor. Precision score is used to measure the number of events identified that are correct. Decision Tree classifier achieves higher precision score compare to both Naïve Bayes and Maximum Entropy classifiers. In fact the Naïve Bayes precision score is very poor when compared to the other two classifiers. Recall score is used to measure the number of correct events that are identified. Naïve Bayes achieves higher recall score when compared to both Maximum Entropy and Decision Tree classifiers.

Overall the Decision Tree classifier achieves the higher F-score compare to both Maximum Entropy and Naïve Bayes classifiers. The Naïve Bayes classifier results are poor compare to both other classifiers in terms of precision and F-score. The results produced by the Decision tree classifier are even slightly better than the one produced by the Maximum entropy classifiers in terms of precision, recall and F-score.

Relations classification results obtained in Table 1 are much better than the results obtained by the [6] by using the biomedical text. One reason for better performance can be that nature of biomedical text is quite complex when compared to MUC-6 data.

Having identified all the pairs of related entities present in any event of MUC-6 data, the next stage is the reconstruction of the complex relations from these

[1] http://mallet.cs.umass.edu/index.php/Main_Page
[2] http://www.english.bham.ac.uk/staff/omason/software/qtag.html

pairs. In order to do this we will first create a graph where an edge between the two entities only exists if the classifier believes that there is a positive relation between them. We just create a simple graph, where entities are represented as vertexes of the graph and edges will represent the positive relation between two entities.

In order to find the maximal clique the [1] algorithm is employed. This algorithm takes a simple graph and returns the largest maximal clique present in the graph.

Events evaluation is performed on the events present in the MUC-6 data. The events are labelled as "*Matched*" if the maximal clique return by the [1] algorithm is same as the events present between *@@TAGS Succession* and *@@ENDTAGS* in the MUC-6 data. Table 2 shows the results of event classification.

The experiment results in Table 1 and Table 2 clearly show that the Naïve Bayes classifier performs very poorly both in binary relations classification and the events evaluation in terms of precision, recall and F-score.

Surprisingly the results produced by the Decision Tree classifier in both binary relations classification and events evaluation are slightly better than Maximum entropy classifier in terms of precision, accuracy and F-score. Maximum Entropy classifier is used by [6] while conducting their experiments on biomedical corpus. Moreover, the feature set used in this paper do not contain the last feature used by [6], as it did not appear to make any difference. The experimental data used by [6] is based on selected abstracts from MEDLINE, as MEDLINE is constantly growing, so it is quite difficult to replicate the same experimental data used by [6].

The results obtained in this paper clearly show that Decision Tree classifier performs better compared to the Maximum Entropy classifier.

The reason behind the better performance of the Decision Tree classifier is that it is well-equipped to deal with the problems where instances are described by a fixed set of attributes. Decision Tree learning methods are also quite robust and efficient and even give good performance when some training examples have unknown values.

Classifiers	**Training Data Accuracy**	**Testing Data Accuracy**	**Precision**	**Recall**	**F-Measure**
Maximum Entropy	0.94127	0.82682	0.9357	0.8663	0.8996
Naïve Bayes	0.84248	0.64980	0.6364	0.9161	0.7510
Decision Tree	0.8394	0.8493	0.9562	0.8740	0.9132

Table 1: Relations Classification Results

Classifiers	**Precision**	**Recall**	**F-Measure**
Maximum Entropy	0.6176	0.4921	0.5477
Naïve Bayes	0.3686	0.7826	0.5011
Decision Tree	0.7578	0.5625	0.6456

Table 2: Results of Events Evaluation

6. Conclusion and Future Work

Information Extraction (IE) is a rapidly growing area of Natural Language Processing. The objective of information extraction is to build systems which find and link relevant information from unstructured text ignoring irrelevant information.

This paper has presented an approach for complex relations extraction in which the complex relations are first factorised into binary relations then different classifiers (Maximum Entropy, Naïve Bayes and Decision Tree) are trained to learn to identify binary relations. In the second phase, complex relations are reconstructed by finding maximal cliques in graphs that represent relations between pairs of entities. At the end the results produced by these different classifiers are compared.

Decision Tree classifier outperforms both Naïve Bayes and Maximum Entropy classifier in terms of precision, recall and F-score. Results produced by the Naïve Bayes classifier are relatively quite poor compared to Maximum Entropy and Decision Tree classifier. The principal benefit of factorising complex relations into binary relation is that it allows the use of any binary relations classifiers which have been well studied and frequently produce accurate results.

For future work, in this paper we have looked at the modified version of MUC-6 data in which events are completely described within a single sentence. It will be interesting to investigate the events described in multiple sentences [8].

Moreover, this approach can be improved by using much deeper synthetic parsing and more powerful binary classifiers based on tree kernels [10]. This approach can also be employed on diverse domains.

At the moment, the presented approach is using supervised learning algorithms and it would be interesting to investigate how this approach will perform when unsupervised learning algorithms are used.

7. References

[1] Bron C. and Kerbosch J. "Algorithm 457: Finding all cliques of an undirected graph". *Communication of the ACM*, 1973 16(9): 575-577.

[2] Chieu H. L., Ng H. T. "A Maximum Entropy Approach to Information Extraction from Semi Structured and Free Text". *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI 2002)* 2002.

[3] Grishman R. "Information extraction: Techniques and challenges". In *Information Extraction (International Summer School SCIE-97)*. Springer-Verlag 1997.

[4] Grishman R., and Sundheim B. "Message Understanding Conference - 6: A Brief History" Proceedings of the 16th International Conference on Computational Linguistics (COLING-96), Copenhagen, Denmark, 1996, pp466-470.

[5] McCallum A. K. "MALLET: A machine learning for language toolkit" 2002.

[6] McDonald R., Pereira F., Kulick S., Winters S., Jin Y. and White P. "Simple Algorithms for Complex Relation Extraction with Applications to Biomedical Text". *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics.* Ann Arbor, Michigan ,2005 Pages: 491 – 498.

[7] Soderland S. "Learning Information Extraction Rules for Semi-structured and Free Text Machine Learning*"*.1999, 34(1):233–272.

[8] Stevenson M. "Information Extraction from Single and Multiple Sentences". *In Proceedings of the Twentieth International Conference on Computational Linguistics (COLING 2004)*, 2004, Geneva Switzerland.

[10] Zelenko D., Aone C., and Richardella A. "Kernel methods for relation extraction". JMLR 2003.