Automatic Language-Independent Induction of Gazetteer Lists

Diana Maynard, Kalina Bontcheva, Hamish Cunningham

Dept of Computer Science, University of Sheffield, Sheffield, UK {diana, kalina, hamish}@dcs.shef.ac.uk

Abstract

Adaptation of existing Information Extraction (IE) systems to new languages and domains is the focus of much current research, but progress is often hindered by the lack of available resources to enable developers to get a new system up and running fast. It has previously been shown that a good set of gazetteer lists can have a vital role here, but creation of lists for a new language or domain can be time-consuming and laborious. In this paper we demonstrate a tool for inducing gazetteer lists from a small set of annotated corpora and creating a baseline IE system. We also describe an extension to this, using bootstrapping techniques in order to generate much larger volumes of noisy training texts. High quality results have been achieved in this way on Hindi, Chinese and Arabic.

1. Introduction

One of the major bottlenecks in adapting IE systems to new languages is the collection and organisation of new lexical resources. For some languages there is a large amount of information available – usually in electronic form such as on the Internet – but for other languages there is very little information available. IE systems fall into two main categories – knowledge engineering approaches which typically use rule-driven systems, e.g. (Maynard et al., 2003a), and machine learning approaches, such as (Bikel et al., 1999). Both approaches typically make use of large gazetteer lists to aid named entity recognition (NE), although rule-based systems are generally more dependent on these than machine learning systems. Such lists contain not only geographical references such as names of cities, countries, etc, but also names of people (especially first names), large organisations, months of the year, days of the week, numbers, etc.

Previous experiments with other languages have shown that good gazetteer lists are one of the keys to success for a rule-based NE system, particularly in the short term (Maynard et al., 2003b) and for recognition of locations (Mikheev et al., 1999). By this we mean that good baseline scores can be achieved with nothing more than a very basic set of components and a comprehensive gazetteer, particularly in terms of Recall. Depending on the language, however, precision may suffer if more sophisticated methods are not used, for example in languages such as Chinese where names of Persons and Organisations are highly ambiguous.

There are several important uses for the gazetteer induction method we describe in this paper.

- It enables rapid creation of gazetteer lists from training data, rather than the often time-consuming process of searching the web for relevant lists.
- It enables rapid creation of a baseline NE system against which other methods can be tested and evaluated. Experiments have shown that the reuse of named entities actually occurs extremely frequently, especially in texts belonging to the same domain and type (for example, news articles from the same source), so that a good baseline can be achieved by using just a set of lists and associated grammars (Palmer and Day, 1997).
- It provides a method for generating noisy training data from a small seed corpus (cf (Morgan et al., 2003)).
- It enables the assessment of ambiguity levels for different entity types in a language. This can be very useful when deciding on a strategy to use for NE recognition (or other procedures), or more specifically, when determining e.g. hand-coded rules for semantic grammar development.
- It is also important for benchmarking purposes, since any evaluation needs to take account of the level of difficulty of the task (disambiguating such entity types), in order to provide a useful result.

2. The Gazetteer List Collector

As part of our work on improving language agility for IE, we created a ”gazetteer list collector”, which will be made freely available within GATE (Cunningham et al., 2002). This tool collects occurrences of entities directly from a small set of annotated training texts, and populates gazetteer lists with the entities. The entity types and structure of the gazetteer lists are defined as necessary by the user. Once the lists have been collected, a semantic grammar can be used to find the same entities in new texts.

The list collector also has a facility to split the Person names that it collects into their individual tokens, so that it adds both the entire name to the list, and adds each of the tokens to the list (i.e. each of the first names, and the surname) as a separate entry. When the grammar annotates Persons, it can require them to be at least 2 tokens or 2 consecutive Person Lookups. In this way, new Person names can be recognised by combining a known first name with a known surname, even if they were not in the training corpus. Where only a single token is found that matches, an Unknown entity is generated, which can later be matched with an existing longer name via the orthomatcher component which performs orthographic coreference between named entities. This same procedure can also be used for other entity types. For example, parts of Organisation names can be combined together in different ways.

3. Using contextual information to bootstrap the lists

The list collector can also be combined with a semantic tagger and used to generate context words automatically.

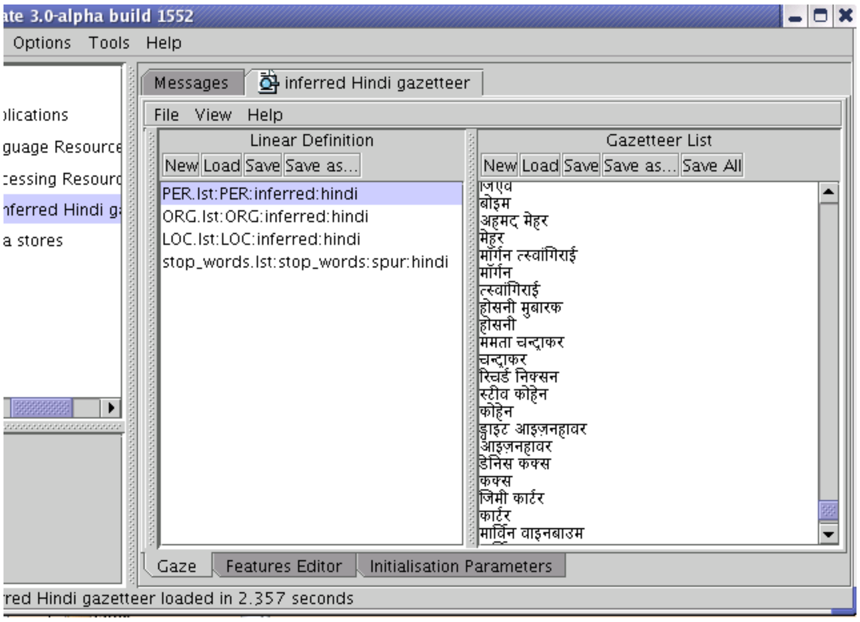

Figure 1: Lists collected automatically for Hindi

Suppose we generate a list of Persons occurring in our training corpus. Some of these Persons will be ambiguous, either with other entity types or even with non-entities, especially in languages such as Chinese. One way to improve Precision without sacrificing Recall is to use the lists collector to identify from the training corpus a list of e.g. verbs which typically precede or follow Persons. The list can also be generated in such a way that only verbs with a frequency above a certain threshold will be collected, e.g. verbs which occur less than 3 times with a Person could be discarded.

4. Using the lists collector to identify relevant contextual information

The lists collector can also be used to improve recognition of entities by enabling us to add constraints about contextual information that precedes or follows candidate entities. This enables us to recognise new entities in the texts, and forms part of a development cycle, in that we can then add such entries to the gazetteer lists, and so on. In this way, noisy training data can be rapidly created from a small seed corpus, without requiring a large amount of annotated data initially.

Using simple grammar rules, we can collect not only examples of entities from the training corpus, but also information such as the syntactic categories of the preceding and following context words. Analysis of such categories can help us to write better patterns for recognising entities. For example, using the lists collector we find that definite and indefinite articles are very unlikely to precede Person entities, so we can use this information to write a rule stipulating that if an article is found preceding a candidate Person, that candidate is unlikely to be a valid Person. We can also use lexical information, by collecting examples of verbs which typically follow a Person entity. If such a verb is found following a candidate Person, this increases the likelihood that such a candidate is valid, and we can assign a higher priority to such a candidate than one which does not have such context.

Below we give a more detailed example of the procedure used as part of the Chinese IE system we created.

4.1. NounPerson collector

The NounPerson collector collects examples of nouns which precede and follow Person annotations in the training corpus. A grammar first identifies nouns immediately preceding and following Person annotations, and annotates such nouns as NounPerson and PersonNoun respectively. Such annotations are placed in a new annotation set called Collected.

The grammar is followed by a gazetteer lists collector which matches the annotation types NounPerson and PersonNoun from the Collected set, and populates a gazetteer called NounPerson_lists.def, consisting of NounPerson.lst and PersonNoun.lst, with the respective relevant strings.

Once these lists have been populated, a second grammar is used to match patterns of (for example) the form "?Person + NounPerson", where "?Person" is a candidate Person and "NounPerson" is an element from NounPerson.lst, i.e. it has been found preceding other Persons in the training corpus. If such a match is found, the candidate Person is annotated as a Person.

A second stage of the process is to place further restrictions on these patterns by only permitting PersonNouns to be added to the list if they occur with a frequency $>$ x (where x is e.g. 2 or 3, depending on corpus size etc.) The size of x should be determined heuristically. This parameter is set in the listscollector.java file, though it would ideally be better to be able to set this as a runtime parameter of the lists collector, so that it could be changed on the fly.

The cycle can be reiterated by automatically adding the new Persons found in this way to the lists, using the Person lists collector, and the process can be repeated.

In addition to collecting nouns in the context of Persons, we also collected verbs occurring before and after Persons, and created appropriate rules and lists for VerbPerson and PersonVerb. Figure 2 shows some examples from the PersonVerb list.

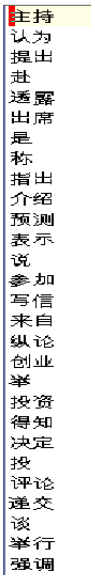

Figure 2: Samples from VerbPerson list collected for Chinese

5. Evaluation

We have tested these methods on Hindi, Chinese and Arabic texts, with great success. We created an NE system for Hindi from scratch in less than one person-month, using little more than a set of gazetteer lists created from training data and a very simple semantic tagger, and achieved an Fmeasure of 71% on news texts. Figure 1 shows a screenshot of one of the Hindi lists collected. This work was part of a US program to develop language processing tools and resources for an unknown language in a very restricted time span, not only for Information Extraction but also for Machine Translation, Cross Language Information Retrieval, etc. Most systems used purely machine learning techniques, which all required the presence of large scale lists (May et al., 2003; Li and McCallum, 2003).

We also created similar baseline systems for Chinese and Arabic, with an F measure of 50% for Chinese and 69% for Arabic using just the gazetteer lists and simple grammars. For Chinese and Arabic, we then improved the systems by adding manually created lists, creating additional tagging rules and incorporating part-of-speech information.

For Chinese, we also implemented the context collector, as described earlier. Adding information about verbs preceding and following Persons, we improved the Fmeasure from 39% to 50%, and further experiments using information about Adjectives (using the same method) improved Precision by 14% without degrading Recall.

6. Extension to the semantic web

The gazetteer list collector is currently used only for populating a flat structure of gazetteer lists, as is typically used for IE systems. The advent of tools and resources for the semantic web brings new challenges to the field of IE, and in particular with respect to Ontology-Based IE (OBIE). The important difference between traditional IE and OBIE is the use of a hierarchical gazetteer structure (i.e. an ontology) instead of the traditional flat structure.

OBIE poses two main challenges:

- the automatic population of ontologies with instances in the text
- the identification of instances from the ontology in the text

6.1. Automatic ontology population

The automatic population of ontologies with instances from the text requires the existence of an ontology and a corpus. From this, an OBIE application identifies instances in the text belonging to concepts in the ontology, and adds these instances to the ontology in the correct location. It is important to note that instances may appear in more than one location in the ontology, because of the multidimensional nature of many ontologies and/or ambiguity which cannot or should not be resolved at this level (see e.g. (Felber, 1984; Bowker, 1995) for a discussion).The gazetteer list collector currently populates flat gazetteer lists. However, hierarchical lists can also be populated in exactly the same way, since the ontology management system in GATE enables this, using a definition mapping file which automatically takes care of the associations of the instances to concepts in the text..

Figure 3 shows a screenshot of the ontology management system in GATE, displayed here with examples from the employment domain. Using the gazetteer list collector would be extremely useful in kickstarting the process of automatic ontology population, for exampe as part of a semantic web application that enables users to create, modify and/or populate their own ontologies from webpages. A user might pre-select concepts in which they are interested, and start selecting instances associated with each concept that they find in the text. The list collector enables the population of the ontology with such instances, and paves the way for either a machine learning OBIE application to take over the population task automatically (by providing training data) or for a rule-based OBIE application to be developed for that ontology and domain.

6.2. Identification of instances from the ontology

Similarly, the gazetteer list collector can help to provide training data and/or produce a baseline OBIE system to identify instances from the ontology in the text (in the same kind of ways as for traditional IE systems). Collecting training data for building OBIE systems for semantic web applications is likely to be a large bottleneck, because very few such systems currently exist and new training data needs to be created from scratch, unlike traditional IE systems for which training data exists in domains like news texts in plentiful form, thanks to efforts from MUC, ACE and other collaborative and/or competitive programs.

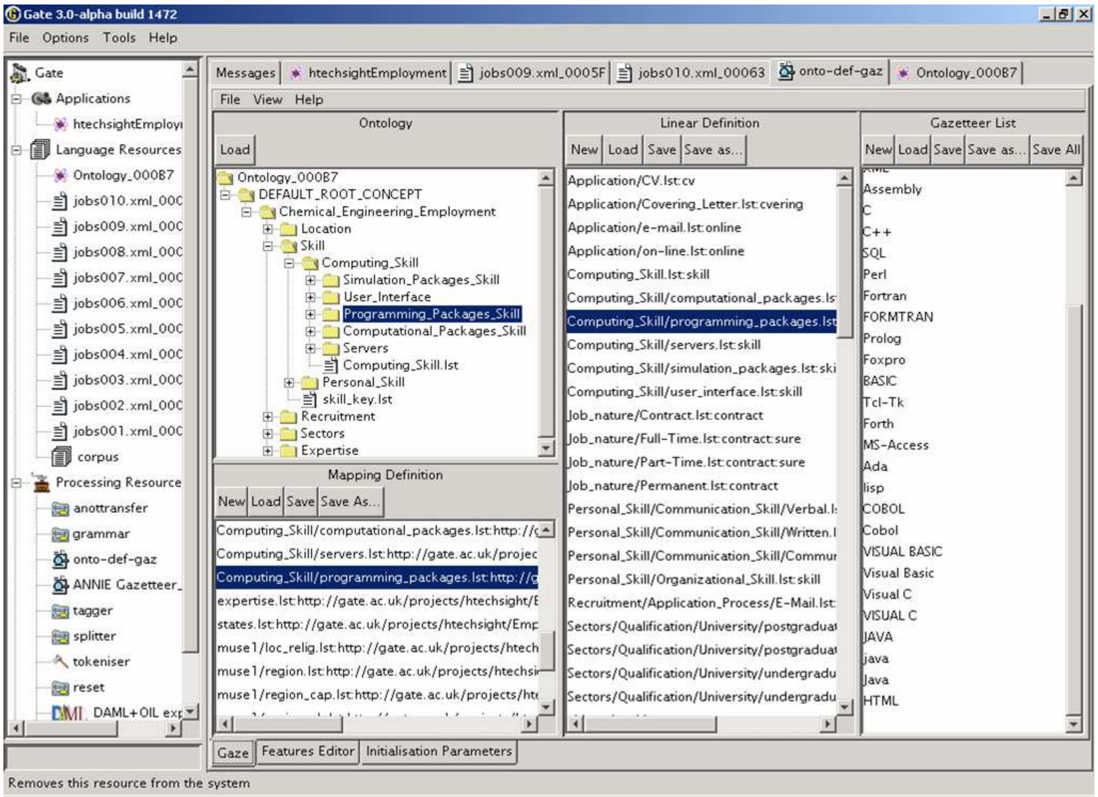

Figure 3: Screenshot of ontology management system in GATE

7. Conclusions

Experiments with automatic gazetteer induction, and in particular with the context creation, are ongoing, but initial results are extremely encouraging, and we have clearly demonstrated the usefulness and feasibility of such methods. In particular, their integration into GATE has improved the language agility and vastly decreased the time taken for adaptation in this respect, without requiring large amounts of annotated training data.

8. References

Bikel, D., R. Schwartz, and R.M. Weischedel, 1999. An Algorithm that Learns What's in a Name. *Machine Learning, Special Issue on Natural Language Learning*, 34(1-3).

Bowker, L., 1995. *A multidimensional approach to classification in Terminology: working with a computational framework*. Ph.D. thesis, University of Manchester, UK.

Cunningham, H., D. Maynard, K. Bontcheva, and V. Tablan, 2002. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*.

Felber, H., 1984. *Terminology Manual*. Paris: Unesco and Infoterm.

Li, W. and A. McCallum, 2003. Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction. *Special issue of ACM Transactions on Asian Language Information Processing: Rapid Development of Language Capabilities: The Surprise Languages*.

May, J., A. Brunstein, P. Natarajan, and R. Weischedel, 2003. Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction. *Special issue of ACM Transactions on Asian Language Information Processing: Rapid Development of Language Capabilities: The Surprise Languages*.

Maynard, D., K. Bontcheva, and H. Cunningham, 2003a. Towards a semantic extraction of Named Entities. In *Recent Advances in Natural Language Processing*. Bulgaria.

Maynard, D., V. Tablan, and H. Cunningham, 2003b. NE recognition without training data on a language you don't speak. In *ACL Workshop on Multilingual and Mixed-language Named Entity Recognition: Combining Statistical and Symbolic Models*. Sapporo, Japan.

Mikheev, A., M. Moens, and C. Grover, 1999. Named Entity Recognition without Gazetteers. In *Proceedings of EACL*. Bergen, Norway.

Morgan, A., L. Hirschman, A. Yeh, and M. Colosimo, 2003. Gene Name Extraction Using FlyBase Resources. In *Proc. of ACL 2003 Workshop on Natural Language Processing in Biomedicine*. Sapporo, Japan.

Palmer, D. and D. Day, 1997. A statistical profile of the named entity task. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP)*. Washington, D.C.