Automatic Discovery of NLP Resources on the Web

Viktor Pekar *(v.pekar@wlv.ac.uk)*
CLG, University of Wolverhampton
Richard Evans *(r.j.evans@wlv.ac.uk)*
CLG, University of Wolverhampton

The World-Wide Web has become a popular vehicle for disseminating and obtaining research and educational resources. In Natural Language Processing (NLP), for example, the specialist community has accumulated a large amount of NLP resources which it has developed over years, including software (part-of-speech taggers, parsers, various corpus analysis tools) and data (evaluation corpora and datasets, frequency lists, glossaries, gazetteers). Most of these resources are freely available, which helps other researchers to save effort in developing them and allows for direct comparisons with previous work. However, finding these resources on the web is not straightforward. Traditional keyword search is often too costly in terms of time and effort. One can look up collections of web links on the topic, such as the ACL/NLP Universe. Unfortunately, as they are manually compiled and maintained, these collections quickly fall out of date and have limited coverage.

The purpose of our work (ESRC grant RES-000-23-0010) is to design a system which will mine the Internet for pages on NLP resources, extract relevant facts from them and produce a publicly accessible database with this information. The task needs to be addressed by a combination of technologies from the areas of language processing and information retrieval. While these technologies have long been investigated in isolation, recent research started to focus on integrating them in complex information systems to be deployed for real-world tasks (e.g., Petasis 2003). The main contribution of this paper is a range of solutions we develop for effective and efficient integration of the technologies. We here present the design of the system under development, focusing on interoperability of its components.

1. Domain crawler

Initial inspection of existing web resources showed there are two most useful sources for their discovery, manually prepared collections of links and email announcements on specialized lists. To find these pages, a web crawler was implemented which (1) seeks out large collections of URLs of interest on the web and (2) selectively downloads pages mentioned in them that may be relevant to the domain. This first step produces an intermediate corpus of potentially useful documents.

2. Format normalization

All the retrieved documents come from extremely diverse sources. For the purpose of further processing, their formatting has to be made uniform. To achieve this, the following pre-processing was performed:

(a) Since email messages typically appear as plain text, the text layout in them (headings, subheadings, itemized lists, emphasized text, etc) was automatically recognized and corresponding HTML tags were inserted.

(b) Character encoding was normalised to ensure that it is uniform throughout the corpus.

(c) HTML errors were detected and corrected using the HTML Tidy tool.

3. Text filtering

To identify relevant documents among those downloaded, the text filtering component implements an interface to *Rainbow*, a freely available text categorization toolkit. It classifies all the downloaded documents into two categories: relevant and irrelevant ones. As training data, the system uses around 100 pages describing various NLP resources and around 900 irrelevant pages, randomly picked from the output of the crawler at a pilot run and manually classified. After testing a range of parameters in the standard categorization procedure (learning algorithm, feature selection parameters, various tokenization options, etc.), the most optimal categorization scheme was determined (F-measure=0.97).

4. Term extraction and gazetteer acquisition

The term extraction component is responsible for the acquisition of the most important terms and named entities (person and organization names, dates, names of software and data resources) in the domain. The list of terms it produces is used to improve the tokenization necessary for text categorization and to construct domain gazetteers. This component looks for the most frequent words and word sequences in the domain corpus, and applies a range of pattern matching rules to filter out errors and recognize particular semantic types of terms and named entities as well as their abbreviations. The term lists are later revised by an expert to ensure their quality.

5. Language identification

One characteristic of the downloaded documents is that many of them contain text in two or more languages (e.g., the description of resources for languages other English). The purpose of the language identification component is to remove non-English paragraphs from a document. It uses a character-based 3-gram model of the language identity of the text, which allows the correct recognition of the language of small text snippets. The component puts special tags around the paragraphs identified as non-English so as to exclude them from further processing.

6. Named entity recognition

The Named Entity (NE) recognition component identifies various semantic types of proper nouns, a step preceding information extraction. NE recognition is carried out by a combination of methods, which include:

(a) Common NEs (person names, locations, and dates) are recognized using the GATE system (Cunningham et al. 2002). Its output is customized to the application domain with the help of a set of post-processing rules (e.g., geographical NEs such as oceans and mountain ranges are irrelevant for the domain, so NE tags are removed from such words).

(b) NEs specific to the domain are further recognized using (1) gazetteers automatically acquired from the domain corpus, (2) a set of transducers (rules for semantic annotation of text).

(c) A newly proposed method, which exploits text layout to learn NER rules from already annotated NEs, was applied to improve the coverage of the NER component.

The NER component is run before language identification and text categorization. Removing all proper nouns from text before recognizing its language helps to reduce errors caused by the presence of foreign names in it. Text categorization profits from substituting unique proper nouns by their semantic category labels, whereby all semantically similar NEs are mapped to the same feature.

7. Coreference resolution

Given that in a particular document, there may be different ways to refer to NEs (e.g., full names, abbreviations, definite descriptions), it is important to link such variants into coreference chains. The component creates coreference chains for the NE types recognized by the NER component by applying a set of pre-defined rules firing orthographic cues (Bontcheva et al. 2002).

8. Information extraction

The information extraction (IE) task consists of identifying relations between recognized NEs, which are later used to fill complex templates (Table 1 describes the template to be filled along with the example fillers). Previous research has developed a range of IE methods for tasks that require filling one template per document (e.g., Kushmerick et al. 1997, Freitag 1998). Here, we opt for an IE method similar to the one proposed by (De Sitter & Daelemans 2003). This method learns two distinct machine learning classifiers from an annotated corpus: one operating on the level of sentences and one on the level of words. First, the sentence-level classifier scans the document for sentences that potentially contain template fillers. After that, the word-level classifier attempts to precisely pinpoint the filler instance in the relevant sentences by looking at the local context of each of its words. Thereby, the context of a word occurrence is represented through features corresponding actual text tokens, all semantic and HTML tags appended on them, and part-of-speech tags.

Field	Example filler
Name	CLAWS part-of-speech tagger
Area	PoS tagging
Creator	UCREL
Licence	Commercial, in-house service
Platform	UNIX
Prog_language	
Req_applications	
Nat_language	English
URL	<http://www.comp.lancs.ac.uk/ucrel/claws/>

Table 1

One difficulty with applying IE to the domain corpus is that very often a particular document does not contain information about all the fields of the template (e.g., the field for natural language remains unfilled for language-independent tools like annotation software). They should be distinguished from documents which do not fill the template fields because they have been erroneously classified as relevant at the text filtering stage. To draw this distinction, a special verification step is applied. It is based on estimating the probability of every field being filled for relevant and irrelevant documents in the gold standard corpus and comparing the corresponding probability vectors with a similar vector prepared for each newly processed document.

A prototype incorporating these components has been implemented and in subsequent work, we will continue to

perform user-focused tuning of this tool to enhance it with a view to deployment in the research domain. Although the initial stages of the BiRD project have addressed IE in the field of computational linguistics, it will be interesting to evaluate the system in application to new domains.

Bibliography

Bontcheva, K., M. Dimitrov, D. Maynard, V. Tablan, and H. Cunningham. "Shallow Methods for Named Entity Coreference Resolution." *Chaînes de références et résolveurs d'anaphores, workshop TALN'2002* (2002).

Cunningham, H., D. Maynard, K. Bontcheva, and V. Tablan. "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications." *Proceedings of ACL'02* (2002).

De Sitter, A., and W. Daelmans. "Information extraction via double classification." *Proceedings of the ECML/PKDD'03 Workshop on Adaptive Text Extraction and Mining (ATEM'03)* (2003).

Freitag, D. "Information Extraction From HTML: Application of a General Learning Approach." *Proceedings of AAAI'98* (1998).

Kushmerick, N., D. Weld, and R. Doorenbos. "Wrapper Induction for Information Extraction." *Proceedings of IJCAI'97* (1997): 729–737.

Petasis, G., V. Karkaletsis, G. Paliouras, I. Androutsopoulos, and C.D. Spyropoulos. "Ellogon: A new text engineering platform." *Proceedings of LREC-02* (2002).