Espotter:

Adaptive Named Entity Recognition for Web Browsing

Tech Report kmi-04-12

Jianhan Zhu, Victoria Uren and Enrico Motta

ESpotter: Adaptive Named Entity Recognition for Web Browsing

Jianhan Zhu, Victoria Uren, Enrico Motta
Knowledge Media Institute, The Open University
Walton Hall, Milton Keynes, MK7 6AA, UK
{j.zhu, v.s.uren, e.motta} @ open.ac.uk

ABSTRACT

Web users are facing information overload problems, i.e., it is hard for them to find desired information on the web. Hence the growing interest in named entity recognition (NER) for discovering relevant information on users' behalf. We present a browser plug-in called ESpotter which adapts lexicons and patterns to a domain hierarchy consisting of domains on the web and user preferences for accurate and efficient NER. Mappings are created from domain independent types to domain specific types. Entities are highlighted according to their types, and users are assisted by navigational functionalities between these highlighted entities.

Categories and Subject Descriptors

I.2.7 [Artificial Intelligence]: Natural Language Processing—text analysis; H.5.4 [Information Interfaces and Presentation]: Hypertext/Hypermedia—navigation

General Terms

Algorithms, Human Factors

Keywords

Web user browsing, Named entity recognition, Hierarchies

1. INTRODUCTION

Users often have problems finding valuable knowledge buried under the overwhelming amount of information on the Web. The sheer amount of information on the Web has made automated methods for discovering these knowledge nuggets on the user's behalf a necessity. In addition, Web pages often contain valuable structured information that is hidden in patterns exhibited in Web page layouts and the regularities of human language. Knowledge in the form of lexicons (such as extracted from dictionaries) can help extract information from these Web pages. Named entity recognition (NER) tackles the problem of finding proper names of various types (such as "John Smith" as a "Person" type, and "Open University" as an "Organization" type) as knowledge nuggets useful to users. NER relies on lexicon matching and pattern matching to find these entities and their types. A lexicon consists of the plain content of an entity and its type (such as "Open University" is an "Organization"). A pattern consists of the formal description of the content structure of a type of entities, which is usually described in regular expressions, and the type (such as a word starting with capitalized letter followed by "University" is an "Organization"). Given the content of a Web page, these lexicons and patterns are sorted in the order of their relevance to the Web page and applied to the content in a sequential order in order to get a list of entities and their types.

2. ESPOTTER

We present a novel NER browser plug-in called ESpotter (Figure 1), which differs from previous NER systems in two aspects. First, lexicons and patterns are adapted to domains on the web. Second, lexicons and patterns are adapted to users.

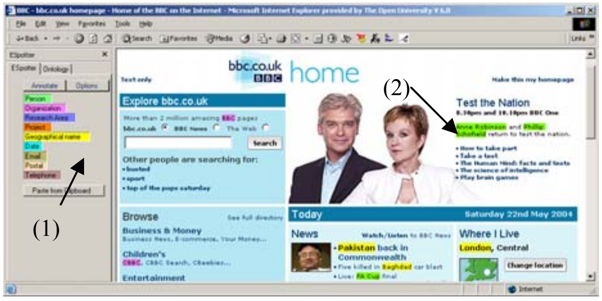

Figure 1. ESpotter highlighting a page on the BBC Website, showing (1) the ESpotter toolbar (2) entities highlighted by ESpotter.

2.1 Domain Adaptation

There are a very large number of lexicons and patterns which can be used for entity recognition. Imagine the number of English names and organization names in the world. Previous NER systems work well on a given domain, but break easily when the content of a document falls outside the domain. On the Web one domain can be only one click away from another. Therefore, in order to support user Web browsing, a NER system needs to be able to adapt to various domains.

Domain adaptation in ESpotter tackles two types of behavior of entities on the Web. First, the same entity means different types of things on different domains. For example, "Magpie" is a type of "Bird" on the Royal Society for Protection of Birds (RSPB) Website, but a type of "Project" on the Knowledge Media Institute (KMi) Website. Second, some entities mostly appear on a certain domain and are not likely to appear on the other domains. For example, UK postal addresses are not likely to appear on Websites other than UK ones.

A domain hierarchy (Figure 2) is used in ESpotter for domain adaptation. A domain hierarchy consists of domains on different levels and links between domains on two adjacent levels. Domains are represented by their URIs (Unified Resource Identifiers). The Root node is on the zero-level. The top level domains are on the first level, such as "uk" and "com". Second level domains are on the second level, such as "ac.uk" and "microsoft.com", and so on. The higher the level of a domain is, the more general the domain is. A link from domain A to domain B means that A is the parent of B.

In ESpotter, a lexicon or pattern is defined on the domain hierarchy and is given a 'precision' between 0 and 1 on each domain. The higher the 'precision' is, the more accurate it is expected to be for that domain.

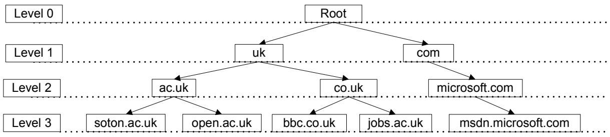

Figure 2. A domain hierarchy defined on domain URIs.

These 'precisions' can be manually assigned by users. Since manual assignment of a very large number of lexicons and patterns on a number of domains is not feasible, we propose a search engine based precision assessment method. A set of search queries, which are composed using the content of a lexicon or the content structure description of a pattern and a domain, D, are used to search Google to get the number of occurrences of matching entities on the domain, Num(total). Each of these search queries are joined with the type of the lexicon or pattern to get the number of occurrences of entities of the type on the domain, Num(type). We divide Num(type) by Num(total) as the 'precision' of the lexicon or pattern on the domain, i.e., Pr(D, type)=Num(type)/Num(total).

2.2 User Adaptation

There are two main aspects to user adaptation. First, since no NER system can have a complete repository of all lexicons and patterns, users are given the opportunity to add their own knowledge. In ESpotter, users can add new lexicons and patterns and assign their precisions on different domains. Second, users can customize ESpotter to fulfil their task at hand. Users can set a 'precision' threshold to adjust the precision and recall of entity recognition, i.e., the higher the threshold, the more accurate NER is, but it may miss some entities. Users can select the types of entities they want to find on a page. They can adjust 'precisions' of lexicons and patterns on different domains. They can modify or disable current lexicons and patterns. They can give feedback on recognition results, which are used by ESpotter to adjust the 'precisions' of lexicons and patterns, e.g., reduce the 'precision' of a lexicon if user says that it made a mistake in entity recognition.

2.3 Entity Recognition

Given the content and URI of a page, ESpotter finds a list of entities on the page and their types. The page content is pre-processed for text segments separated by HTML tags. Users can specify parts of the page used for NER. Since most entities in English are in the form of proper nouns (PNs) [1, 3], lexicons and patterns are divided into PN dependent and PN independent ones. For each text segment, PN independent lexicons and patterns above the 'precision' threshold are sorted in the order of their relevance decided by their 'precisions' on the domain of the current page, its parent domain, and ancestor domains and applied to find a list of entities and their types. A PN matching pattern is then applied to find a list of PNs. For each PN, PN dependent lexicons and patterns above the 'precision' threshold are similarly sorted in the order of their relevance and applied to find a list of entities and their types.

2.4 Entity Type Mapping

Since lexicons and patterns are used for entity recognition on various domains, they generally find entities of domain independent types (such as "Project"). For example, a lexicon finds "Magpie" as a "Project" on KMi Website. Given the current domain, we can create mappings from these domain independent types to domain specific types (such as "KMi Project"), which are sub-types of these domain independent types. For example, we map "Project" to "KMi Project" on the KMi Website. We use content similarity between entities of domain independent types and entities of domain specific types to further improve the accuracy of these mappings.

2.5 Highlighting for Web User Browsing

Each type is assigned a color, which can be configured by users. ESpotter uses <Span> tags to specify the background colors of recognized entities in the HTML source of the page. ESpotter enables users to switch among highlighted entities and view lists of entities, which are sorted by their types or contents. Users can save highlighted pages and NER results.

2.6 Evaluation

We selected ten Websites and five Web pages from each Website for NER using ESpotter. For ten types of entities (such as "People", "Organization"), ESpotter achieved an average precision 81% and recall 62%. After user customization (such as additions of new lexicons and patterns), the average precision and recall are improved to 92% and 82% respectively.

3. PREVIOUS WORK

KNOWITALL [3] extracts facts, concepts, and their relationships from the web. A template based on language grammars is instantiated to find instances of classes. Perkowitz et al. [4] exploit the web as a repository of human beliefs in order to mine large libraries of human activities from the web. GCPs (Google Conditional Probabilities) are used to measure similarity between concepts and terms. PANKOW [1] employs an unsupervised and pattern-based approach to categorize instances with regard to an ontology of classes. PNs are extracted from web pages as candidate instances. A set of patterns, which identify isa-relationships between an instance and a class, are used to construct search queries to get the number of hits from Google. Magpie [2] uses an ontology to semantically markup web pages on-the-fly. Entities of various ontological classes are highlighted for user browsing.

4. CONCLUSIONS AND FUTURE WORK

ESpotter improves on previous NER tools by adapting lexicons and patterns to Web domains and users. ESpotter URI based approach to domain adaptation helps solve the lexicon and pattern overload problem for accurate and efficient NER. We plan to complement ontology based NER with generic NER by integrating ESpotter with the semantic browsing tool Magpie.

5. ACKNOWLEDGEMENTS

This work was supported in part by the Dot.Kom project under grant IST-2001-34038.

6. REFERENCES

[1] Cimiano, P., Handschuh, S., and Staab, S. (2004) Towards the Self-Annotating Web. *WWW 2004*, New York, USA.

[2] Domingue, J. B. and Dzbor, M. (2004) Magpie: Browsing and Navigating on the Semantic Web. *IUI 2004*, Portugal.

[3] Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D.S., and Yates, A. (2004) Web-scale Information Extraction in KnowItAll. *WWW 2004*, New York, USA.

[4] Perkowitz, M., Philipose, M., Fishkin, K., and Patterson, D. J. (2004) Mining Models of Human Activities from the Web. *WWW 2004*, New York, USA.