# Web Directories as Training Data for Automated Metadata Extraction

Martin Kavalec, Vojtěch Svátek and Petr Strossa

Department of Information and Knowledge Engineering
University of Economics, Prague, 13067 Praha 3, Czech Republic
e-mail: kavalec@vse.cz, svatek@vse.cz, kizips@vse.cz

## 1 Introduction

Although *man-made* annotations are considered as the main 'knowledge fuel' for the Semantic Web, the majority of existing commercial pages are still poorly equipped with any kind of metadata, never mind the forthcoming standards such as the RDF syntax or the Dublin Core semantics. *Information Extraction*, relying on characteristic patterns in text, can be applied even on such 'legacy' pages, in order to obtain metadata containing, for example, the names, types, and domains of activity of the WWW subjects (companies).

The two decades of development of Information Extraction techniques have shown that extraction patterns applicable on real-world, unstructured text data cannot be satisfactorily prepared by hand; instead, Machine Learning (ML) became the enabling technology. The common assumption in ML-based Information Extraction is that the training cases are chunks of text pre-labelled by a human indexer. This is acceptable for domain-specific resources with limited vocabulary and more-or-less conventional structure, such as computer science department pages [2] or housing advertisement pages [7]. However, if we proceed to broad categories such as 'pages of companies offering products or services', the number of training cases needed will explode, the acquisition of text fragments becomes difficult, and their manual labelling simply infeasible. Yet, there is a promising resource of web data that has already undergone a process of human indexing, of a sort: web directories such as Open Directory or Yahoo!

In this paper, we analyse the possibility of reusing the knowledge embedded in the structure of the directories in order to obtain *labelled* training data for Web Information Extraction with limited human effort. In section 2 we show the results of preliminary experiments consisting in mining the fragments of web pages, obtained with the help of web directory information, for indicator terms usable for subsequent extraction of semantic information from other pages. In section 3 we outline an ontology of web directories, and suggest the way it can be used to refine the above process. In section 4, our approach is compared to some other projects. Finally, in section 5, we summarise our plans for the future.

## 2 Mining Indicator Terms through Directory Headings

Our assumption is that the *directory headings* (such as `... /Manufacturing /Materials/Metals/Steel/...`) coincide with the generic names of products and services—let us nickname them *informative terms* in this paper—offered by the owners of the pages referenced by the respective directory page. By matching the headings with the page fulltexts, we obtain sentences that contain the informative terms. The terms situated near the informative terms in the structure of the sentence are candidates for *indicator terms*, provided they occur frequently on pages from various domains. The resulting collection of indicator terms can, conversely, play the role of 'extraction patterns' for discovering informative terms in previously unseen pages.

The knowledge asset embedded in web directories is the judgement of human indexers who have assigned the pages under the particular heading(s). Naturally, informative terms on the page need not always correspond to the existing directory headings, e.g. due to synonymy. As consequence, our method will extract (without the help of a thesaurus) only a fraction of the sentences with informative terms. This however does not disqualify the method, since, in this training phase, we aim at discovering indicator terms rather than at identifying the informative terms themselves. The small degree of completeness of the method is actually compensated by the hugeness of the material available[1] in the directories. Namely, the 'Business' subhierarchy of Open Directory (`www.dmoz.org`), which we have exploited in our experiments, points to approx. 150,000 pages overall, each of these containing the 'heading' terms (from the referencing node or one of its ancestors) in two sentences, on the average.

We have tested the training phase of our method on a sample of 14,500 sentences[2] containing the 'heading' terms. The syntactical analysis has been carried out using the *Link Grammar Parser* [6]. The *verbs* which occurred the closest (in the parse tree) to informative terms have been counted, and arranged into a frequency table. In Table 1, the essence of the table is shown, mostly featuring verbs that are likely to be associated with the informative terms (e.g. 'our assortment *includes*...', 'we *manufacture*...', 'in our shop you can *buy*...'). The table contains only the verbs that occurred in at least 50 sentences[3]. We hope to build a more comprehensive collection using a larger sample of pages. Furthermore, the plain verbs (in particular 'to be', which has no significance of its own) can be extended to more complex *phrases*, again via selecting the neighbouring terms with frequent occurrence.

[1] As we dispense with manual labelling, processing a larger sample of data is merely the matter of computer time/storage.

[2] I.e. about 5% of the total of such sentences.

[3] In this display, they are however not arranged according to the relative frequency of occurrence in the neighbourhood of the informative term ($P_n$), but according to the ratio of this frequency to the relative frequency of occurrence in the whole of the extracted, possibly compound, sentence ($P_s$). This visibly pushes down the universal verbs such as 'to be'.

| $P_n/P_s$ | Verb | $P_n$ | $P_s$ |
|---:|---|---:|---:|
| 2.23 | includes | 0.0048 | 0.0021 |
| 2.20 | manufacture | 0.0038 | 0.0017 |
| 2.17 | buy | 0.0038 | 0.0017 |
| 2.09 | including | 0.0057 | 0.0027 |
| 2.08 | supply | 0.0036 | 0.0175 |
| 1.96 | offers | 0.0119 | 0.0060 |
| 1.92 | provides | 0.0135 | 0.0070 |
| 1.92 | offer | 0.0200 | 0.0104 |
| 1.89 | include | 0.0062 | 0.0032 |
| 1.85 | specializing | 0.0051 | 0.0027 |
| 1.78 | providing | 0.0091 | 0.0051 |
| 1.70 | specializes | 0.0045 | 0.0026 |
| 1.66 | specialize | 0.0053 | 0.0032 |
| 1.56 | using | 0.0037 | 0.0024 |
| 1.55 | provide | 0.0191 | 0.0122 |
| 1.40 | use | 0.0046 | 0.0033 |
| 1.39 | sell | 0.0039 | 0.0028 |
| 1.26 | see | 0.0046 | 0.0036 |
| 1.25 | are | 0.0740 | 0.0589 |
| 1.24 | were | 0.0042 | 0.0034 |
| 1.23 | made | 0.0040 | 0.0032 |
| 1.22 | make | 0.0066 | 0.0054 |
| 1.15 | need | 0.0053 | 0.0046 |
| 1.12 | is | 0.0988 | 0.0880 |
| 1.05 | get | 0.0043 | 0.0041 |
| 1.03 | find | 0.0060 | 0.0058 |
| 1.03 | meet | 0.0043 | 0.0041 |
| 1.00 | related | 0.0035 | 0.0035 |

**Table 1.** Frequent verbs in sentences containing the headings

## 3 Ontology of Web Directory Headings

As we have shown in the previous section, interesting pieces of information can be extracted from web directories even without specific assumptions about the nature of the heading terms. Nevertheless, we believe that only deeper *ontological analysis* of the headings can bring the automated discovery of indicators to its full potential, in particular for complex terms spanning across multiple levels of headings. We will thus now outline an ontology of web directory headings.

The semantic information associated with the particular page, in the context of a web directory, is defined by the sequence (or, several sequences, in the case of a non-tree hierarchy) of headings preceding the node pointing to that page. The headings essentially belong to one of the following classes:

1. 'Entity' terms (most often nouns), which correspond to real-world entities. Note that their meaning may depend on the preceding terms: for example, pages referenced by the node preceded by the subpath `Cranes/Accessories` are likely to offer accessories for cranes but not clothing accessories. Nevertheless, the word 'accessories' can possibly be found on the page even without the attribute 'for cranes', since the latter can be assumed by context.
2. 'Property' terms (most often adjectives), which correspond to properties of entities. They are *restrictive* rather than descriptive, since they (usually) restrict the scope of the immediately preceding 'entity' term to denote a narrower class of entities. For example, the pages referenced by the node preceded by the (sub)path `Telecommunications/Wireless` can be viewed as 'indexed' by the compound term 'wireless telecommunications'. The 'property' term is completely dependent on the given 'entity' term, seeking it independently on a page would be spurious.

The 'entity' terms can be further refined to:

1. *Subjects* (active entities) such as `Manufacturers`, `Publishers`, or `Associations`.
2. *Objects* (passive entities) such as `Materials`, `Aircraft` or `Textiles`.
3. *Domains* of activity such as `Telecommunications` or `Publishing`.

In addition, we can identify a *common* subclass of both *object* and *domain*, which can be denoted as *activity.* An activity is a domain, since it fits into the generality hierarchy of domains, but it is also an object, since it can be viewed as a 'commodity' offered by a certain subject, e.g. `Manufacturing` or `Construction`. Furthermore, a distinct feature of an activity is the aptitude of being *applied* on an (other) object.

The diagram at Fig. 1 depicts the essence of the ontology. Boxes correspond to classes, full edges to named relations, and dashed edges to the class-subclass relationship. Reflexive binary relations are listed inside the respective boxes.

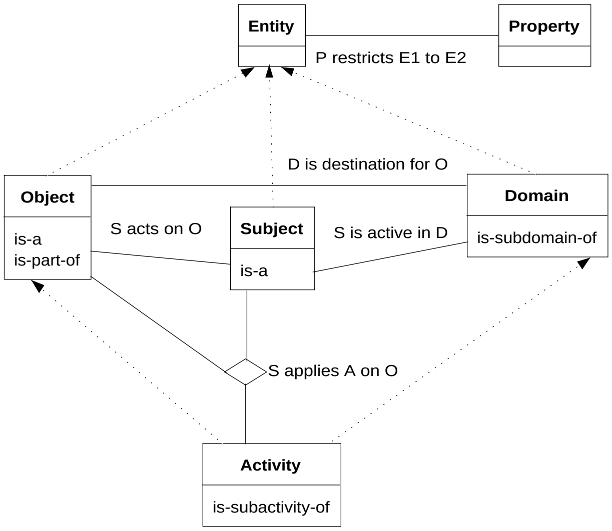

**Fig. 1.** The ontology of web directory headings

Given the ontology, the method described in section 2 can be enhanced in the following way:

1. *Select* the most promising paths and nodes in the directory structure.
2. Assign *class labels* to the headings from the selected paths, and arrange them into a semantic network of *relations.*
3. Generate full-text *queries* based on the headings (and their classes), and apply them on the pages to extract sentences.

4. The mining of indicator terms could then be done *separately* for each class of ‘entity’ terms, thus obtaining collections of ‘extraction patterns’ specific for different type of information to be extracted.

The structure over the headings should enable to generate[4] finer queries, and thus to obtain a more comprehensive sample of training data for indicator learning. As an example, given the path segment `Cranes/Accessories`, in which both `Cranes` and `Accessories` were pre-classified as ‘objects’, and `Cranes` identified as ‘destination’ for `Accessories`, the training cases containing e.g. the expression ’accessories *for* cranes’ might me the most desirable ones.

The importance of step 1 (selection) follows from the fact that step 2 (labelling) has to be done manually, but, in distinction to ‘classical’ labelling, a single labelling action can lead to class (or, relation) assignment to several training cases. The efficiency of manual labelling is thus closely related to the number of pages referenced by the node being labelled, as well as to the number of sentences (from these pages) containing the headings. In order to obtain a high ‘assignments-to-actions’ ratio, we have to trade off the high number of pages (for general headings close to the root, pertaining to huge subhierarchies) with the higher number of sentences per page (for the headings close to the leaves, which are better tuned to the page content, and even subsume several ‘ancestor’ terms). The parameters of the respective utility function could be determined in the future.

## 4 Related Work

The common approach to overcome the lack of classified training examples in text categorisation is to apply *statistical techniques* consisting in iterative automated labelling of unclassified examples based on a few classified ones (bootstrapping, see [1], [4], [5]). So far, we have not considered such techniques, and instead rely on the prior work of a human indexer of the web directory. While directories have already been used for learning to classify *whole documents* [3], their use for *information extraction* seems to be rather innovative.

Our work is actually rather similar to Brin [1], which targets on automated discovery of extraction patterns using *search engines.* The patterns can be used to find relations, such as books, i.e. pairs (author, title). The patterns are based simply on characters surrounding the occurence of investigated relation. In comparison, we aim at finding less structured information, for which such simple patterns wouldn’t be sufficient; we therefore search for linguistic indicators, which are based on syntax analysis. (The indicators themselves can be thought of as ‘syntactic patterns’.)

[4] We are currently working on a rewriting grammar that will automatically convert the set of relational expressions on headings into a layered set of query terms.

## 5 Conclusions and Future Work

We have suggested a novel method for learning *indicative terms*, which can be, in turn, used to extract *important terms* (in fact, meta-data) from web pages. The source of learning cases is a *web directory*: thanks to the prior work of human indexers of the directory, the burden of manual case labelling is either completely removed, or significantly reduced. Preliminary results in a rather restricted setting suggest that the method may be viable.

As we have mentioned in the end of section 2, we will soon extend the non-interactive method of mining the indicators by *searching forth in the parse trees*, beyond the neighbouring verb (in particular for the 'unclear' verbs). The *accuracy on unseen pages* also has to be thoroughly tested. Furthermore, the prospective use of the web directory *ontology* has been described in section 3.

Finally, we anticipate that best results could be obtained by combining our reuse of human effort (with rather precise but incomplete results) with bootstrapping techniques mentioned in section 4 (more complete but possibly imprecise), in a more distant future.

The research has been partially supported by the grant no. 201/00/D045 (Knowledge model construction in connection with text documents) of the Grant Agency of the Czech Republic.

## References

1. Sergey Brin. Extracting Patterns and Relations from the World Wide Web. In: WebDB Workshop at EDBT'98.
2. Dayne Freitag. Information Extraction From HTML: Application of a General Learning Approach. In *Proc. 15th National Conference on Artificial Intelligence (AAAI-98).*
3. Dunja Mladenic. Turning Yahoo into an Automatic Web-Page Classifier. In *Proceedings of the 13th European Conference on Aritficial Intelligence, ECAI'98*, pp. 473-474.
4. Andrew McCallum and Kamal Nigam. Text Classification by Bootstrapping with Keywords, EM and Shrinkage. In *ACL'99 Workshop for Unsupervised Learning in NLP*, 1999.
5. Ellen Riloff and Rosie Jones. Learning Dictionaries of Information Extraction by Multi-Level Bootstrapping. In *Proc. 16th Nat. Conf. Artificial Intelligence (AAAI-99).*
6. Daniel Sleator and Davy Temperley: Parsing English with a Link Grammar. In *Third International Workshop on Parsing Technologies*, August 1993.
7. Stephen Soderland. Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning* Vol. 34, 1999, pp.233-272.