Information extraction from HTML product catalogues: coupling quantitative and knowledge-based approaches

Martin Labský[1], Vojtěch Svátek[1], Pavel Praks[2], Ondřej Šváb[1]

[1] Department of Information and Knowledge Engineering,
University of Economics, Prague, W. Churchill Sq. 4, 130 67 Praha 3, Czech Republic
{labsky,svatek,xsvao06}@vse.cz
[2] Department of Applied Mathematics, Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
pavel.praks@vsb.cz

Abstract. We describe a demo application of information extraction from company websites, focusing on bicycle product offers. A statistical approach to text analysis (Hidden Markov Models) is used in combination with latent semantic analysis of image collections and with ontological knowledge. The results are fed up into an RDF repository and made available through a template-based query interface.

1 Introduction

Tools and techniques for *web information extraction* (WIE) have recently been recognised as one of key enablers for semantic web (SW) scaling. In our long-term project named *Rainbow*[3] we address several intertwined topics that we consider important for efficient 'WIE for SW' applications:

1. Exploitation of *multiple information modalities* available in web documents
2. Synergy of extractor *learning* and reuse of *ontological information*
3. Automated acquisition and labelling of *training data* for extractor learning
4. Bridging between automated *acquisition* of SW data and their *usage*
5. Support for easy design of WIE applications *from components*.

In this paper, we focus on an ongoing demo application in the domain of *bicycle product offers*. Section 2 presents the core method: automated *HTML annotation* based on Hidden Markov Models. Section 3 extends the analysis of HTML code with latent semantic analysis of *images*. Section 4 describes the composition of product offer instances with the help of simple *ontology*. Section 5 outlines the *architecture* of the demo application and the subsequent usage of extracted data in an *RDF repository*. Finally, section 6 focuses on future work.

[3] http://rainbow.vse.cz

2 Web Page Annotation Using HMMs

For extracting product entries from web catalogues, we built a Hidden Markov Model (HMM) tagger, which assigns a semantic tag to each token from a document. In our experiments, we evaluated the HMM performance on a diverse set of web pages, which come from different web sites and have heterogenous formatting.

We manually annotated a set of 100 HTML documents chosen from the Google Directory *Sports-Cycling-BikeShops-Europe-UK-England*. Each document contains from 1 to 50 bicycle offers, and each offer consists of at least the bicycle name and price. There are typically 3–4 documents from the same shop in the data. Annotations for 12 bicycle characteristics were made using SGML tags[4].

HMMs are probabilistic generative models, which represent text as a sequence of tokens. To represent web documents, we employed extensive pre-processing. Similar to [7], we transform each document into XHTML and perform canonicalisation of named entities. Certain HTML tags and tag groups are replaced by their generalisations. Since only words and images can be extracted, we dispose of mark-up blocks that do not directly contain words or images.

The structure of our HMM is inspired by [6]. Slots (i.e. semantic types of information) to be extracted are modelled using *target* states, and accompanied with two types of helper states responsible for representing the slot's characteristic context - the *prefix* and *suffix* states. Irrelevant tokens are modelled by a single *background* state. Contrary to [6] and [16], which use independent HMMs trained for each slot separately, we train a single composite HMM capable of extracting all slots at once. This approach, also used in [1], captures ordering relations between nearby slots (e.g. bicycle image often follows its name). We experimented with different HMM architectures with results presented in [15].

3 Impact of Image Classification

For the purpose of extracting product images, we examined the impact of image information available to the HMM tagger. As a baseline approach, we measured the tagging performance when no image information was available for tagging. In this case, all images where represented by the same token and product pictures could only be distinguished based on the context in which they appeared.

In order to provide our tagger with more information, we built an image classifier to determine whether the extracted product is also depicted as an image. We used the following features for classification: image dimensions, similarity to training product images, and whether there is more than one occurrence of the same image in the containing document. Image *dimensions* appeared to be the best predictor and were modelled using a 2-dimensional normal distribution. *Similarity* was computed using the *latent semantic* approach described in [10]. All three features were combined using rules manually estimated on a separate

[4] The training data are available from `http://rainbow.vse.cz/bikes/`.

held-out set. Using 10-fold cross validation on $1,000$ images from our document collection, the classifier's error rate was 6.6%.

The above binary classifier was then adapted to classify into 3 classes: Pos, Neg, and Unk. Before tagging a document, every image was replaced with one of these 3 classes. In this way, the HMM tagger learned to classify the Pos and Neg classes correspondingly, and the tagging of the Unk class depended more strongly on the context. Note that bicycle images that are not part of product offerings are not tagged for extraction, which increases the role of context for correct tagging. This new image information lead to an increase of 19.1% points in extraction precision for the picture tag and also to subtle improvements for other tags. Changes in precision and recall for 3 chosen slots (product pictures, names and prices), measured on a per-token basis, are shown in Table 1.

Table 1. 10-fold cross-validation results for selected tags over 100 documents

	No image information			Image classes		
Tag	Precision	Recall	F-measure	Precision	Recall	F-measure
Name	83.7	82.5	83.10	83.8	82.5	83.14
Price	83.7	94.4	88.73	84.0	94.4	88.90
Picture	67.8	87.1	76.25	86.9	89.1	87.99

4 Ontology-Based Instance Composition

Semantic web is not about isolated tagged items but about complex and interrelated entities; we thus need to group the labels produced by automated annotation. We currently use a simple sequential algorithm that exploits constraints defined in a tiny *presentation ontology*[5] [9], which partly pertain to the generic domain (bike offers) and partly to the way of presenting information in web catalogues. The constraints mostly deal with multiplicity, uniqueness and optionality of various properties associated with the class 'Bike offer'. An annotated item is added to the currently assembled (bike) instance unless it would cause inconsistency; otherwise, the current instance is saved and a new instance created to accommodate this item and the following ones. Despite acceptable performance on error-free, hand-annotated training data, where the algorithm correctly groups about 90% of names and prices, this 'baseline' approach achieves very poor results on automatically annotated data: on average, less than 50% of corresponding names and prices are matched properly, often for trivial reasons. The most critical problems are connected with *missing slots*, *multiple different references* to a single slot, and with *transposed tables*.

[5] Similar to 'extraction ontologies' used by Embley [5].

5 Result Transformation, Storage And Retrieval

All components developed within the *Rainbow* project are wrapped as web services. The WIE component itself is currently being called by a simple *control routine* (written in Java), which also optionally calls other analysis tools: in the bicycle application, we so far experimented with URL-based navigation over the website, extraction of the content of selected META tags and extraction of 'company profile sentences' from free text[6]. The results are transformed to RDF (with respect to a 'bicycle-offer RDFS ontology') and stored in a *Sesame* [2] repository. An end-user interface to this repository was developed[7]; it relies on a collection of *query templates* expressed in SeRQL (the native query language of *Sesame*) and enables a simple form of navigational retrieval [15].

6 Future Work

Most urgently, we need to replace the 'toy' implementation of ontology-based *instance composition* with a version reasonably robust on automatically annotated data. For some of the *layout-oriented* problems mentioned in section 4, partial solutions recently suggested in IE research (e.g. [3, 5]) could be reused. We also consider extending the reach of *HMMs* even to this phase of extraction; a modified version of Viterbi algorithm supporting domain constraints (such as those in our presentation ontology) has already been described in [1]. Another aspect worth investigation is the possibility of (semi-)automatic construction of presentation ontologies from the corresponding *domain ontologies*.

A critical bottleneck of ML-based IE methods (in particular of statistical ones) is the volume of *labelled training data* required. In our experiments with product catalogues, we noticed that the tagger often classifies most product entries correctly but misses a few product names that are very different from the training data. We developed a simple symbolic algorithm that identifies similar *structural patterns* in a document. For example, the HTML tag sequence `<td> <a> <font> <br/> </font> </a> </td>` with arbitrary words in between appears 34 times in one of our training documents: the tagger successfully annotated 28 product names contained in these patterns between `<font>` and `<br/>`, but missed the remaining 6. In such cases, we could collect the remaining product names and use them to enrich the model's training data. By learning novel product names from these 'easy' pages, the model will learn to also recognise them in less structured documents[8]. We also plan to bootstrap the method with data picked from *public resources* related to product offering, following up with our earlier experiments with Open Directory headings and references [8].

Another important task is to replace hard-coded *control routines* with semi-automatically constructed, implementation-independent application models. A

[6] These three approaches to website analysis, implemented independent of the bicycle demo application, were evaluated in [11].

[7] Available at `http://rainbow.vse.cz:8000/sesame`.

[8] Similar bootstrapping strategies are shown in [4].

knowledge modelling framework has already been introduced for this purpose [13]; currently we examine the adaptability of a PSM-based semantic *web-service configuration* technique in connection with this framework [14].

Eventually, we plan to associate our efforts with the popular *Armadillo* project [3], with which we share most of our abovementioned research interests.

References

1. Borkar V., Deshmukh K., Sarawagi S.: Automatic segmentation of text into structured records. In: SIGMOD Conference, 2001.
2. Broekstra J., Kampman A., van Harmelen F.: Sesame: An Architecture for Storing and Querying RDF and RDF Schema. In: Proc. ISWC 2002, Springer LNCS no. 2342.
3. Ciravegna, F., Chapman, S., Dingli, A., Wilks, Y.: Learning to Harvest Information for the Semantic Web. In: ESWS-04, Heraklion, Springer LNCS 2004.
4. Dingli A., Ciravegna F., Guthrie D., Wilks Y.: Mining Web Sites Using Unsupervised Adaptive Information Extraction. In: EACL, 2003.
5. Embley, D.W., Tao, C., Liddle, S.W.: Automatically extracting ontologically specified data from HTML tables with unknown structure. In: ER2002, Tampere 2002, 322-337.
6. Freitag D., McCallum A.: Information extraction with HMMs and shrinkage. In: Proceedings of the AAAI-99 Workshop on Machine Learning for IE, 1999.
7. Grover C., McDonald S., Gearailt D., Karkaletsis V., Farmakiotou D., Samaritakis G., Petasis G., Pazienza M., Vindigni M., Vichot F., Wolinski F.: Multilingual XML-Based Named Entity Recognition for E-Retail Domains. In: LREC Conference, Las Palmas, 2002.
8. Kavalec, M., Svátek, V.: Information Extraction and Ontology Learning Guided by Web Directory. In: ECAI Workshop on NLP and ML for ontology engineering. Lyon 2002.
9. Labský, M., Svátek, V., Šváb, O.: Types and Roles of Ontologies in Web Information Extraction. In: ECML/PKDD04 Workshop on Knowledge Discovery and Ontologies, Pisa 2004.
10. Praks P., Machala L., Snášel V.: Iris Recognition Using the SVD-Free Latent Semantic Indexing. In: MDM/KDD - Fifth International Workshop on Multimedia Data Mining, Seattle, 2004.
11. Svátek V., Berka, P., Kavalec, M., Kosek, J., Vávra, V.: Discovering Company Descriptions on the Web by Multiway Analysis. In: Intelligent Information Processing and Web Mining, IIPWM'03., Springer Verlag, 2003.
12. Svátek V. et al.: Rainbow - Multiway Semantic Analysis of Websites. In: The 2nd Int. DEXA Workshop on Web Semantics, IEEE Computer Society Press 2003.
13. Svátek, V., Labský, M., Vacura, M.: Knowledge Modelling for Deductive Web Mining. In: Proc. EKAW 2004, Springer Verlag, LNCS, 2004.
14. Svátek, V., ten Teije, A., Vacura, M.: Web Service Composition for Deductive Web Mining: A Knowledge Modelling Approach. In: Proc. Znalosti 2005, VSB-TU Ostrava, to appear 2005.
15. Šváb, O., Labský, M., Svátek, V.: RDF-Based Retrieval of Information Extracted from Web Product Catalogues. In: SIGIR'04 Semantic Web Workshop, Sheffield.
16. Valarakos A., Sigletos G., Karkaletsis V., Paliouras G.: A Methodology for Semantically Annotating a Corpus Using a Domain Ontology and Machine Learning. In: RANLP Conference, Borovets, 2003.