Combining Ontological Knowledge and Wrapper Induction techniques into an e-retail System[1]

Maria Teresa Pazienza, Armando Stellato and Michele Vindigni

Department of Computer Science, Systems and Management, University of Roma Tor Vergata, Italy
{pazienza, stellato, vindigni}@info.uniroma2.it

Abstract. E-commerce and the continuous growth of the WWW has seen the rising of a new generation of e-retail sites. A number of commercial agent-based systems has been developed to help Internet shoppers decide what to buy and where to buy it from. In such systems, ontologies play a crucial role in supporting the exchange of business data, as they provide a formal vocabulary for the information and unify different views of a domain in a shared and safe cognitive approach. In CROSSMARC (a European research project supporting development of an agent-based multilingual/multi-domain system for information extraction (IE) from web pages), a knowledge based approach has been combined with machine learning techniques (in particular, wrapper induction based components) in order to design a robust system for extracting information from relevant web sites. In the ever-changing Web framework this hybrid approach supports adaptivity to new emerging concepts and a certain degree of independence from the specific web-sites considered in the training phase.

1 Introduction

The continuous growth of the Web accesses and e-commerce transactions is producing a new generation of sites: e-retail portals, willing to help end-users in choosing among similar products from different manufacturers, shown in an uniform context (to make easier their comparison). A number of commercial systems are being developed to automatically extract, summarize and show to the end-user relevant data from on-line product descriptions. Most of them neither use natural language technologies nor employ machine learning techniques, being based on shallow approaches that rely on page structure and/or HTML tags to retrieve information of interest. As a consequence, they must be manually tuned on specific pages of monitored sites, and do several assumptions, for example product names, prices, and other features to always appear in a fixed (or at least regular) order, or even pages to be expressed in uniform and monolingual manner (usually English) while it is generally not the case.
Extracting semi structured data from e-retail sites (and in general from the Web) appears to be a complex task, as target information is organized to be appealing and readable by human end-users and not by automatic extraction systems.

Ontologies play a crucial role in supporting information extraction, as they may be considered a formal vocabulary for the information and unify different views of a domain in a safe cognitive approach [9].
We describe here our contribution in building the knowledge base and the IE component as it has been developed inside CROSSMARC, an e-retail product comparison agent system (currently

[1] This work has been partially founded under the CROSSMARC project (IST 2000-25366) of the Information Society Technologies Programme of the European Union.

under development as part of an EU-funded project), where wrapper induction techniques ([1], [2], [4]), are boosted by background knowledge and linguistic analysis both to extend their capabilities and to further ease adaptation to domain changes.
CROSSMARC aims both to stress commercial-strength technologies based on language processing methodologies for information extraction from web pages and to provide automated techniques for an easy customization to new product domains and languages. Its technology currently operates for English, Greek, French, and Italian languages and is being applied to two different product domains: computer goods and job offers: the first one being characterized by brief and semi-structured descriptions, rich of technical terms and acronyms, while the second, IT job offers, contains wider linguistic descriptions. These domains have been chosen to evaluate the system in large on different presentation styles, contents, use of tables and layout aspects.
In the following section, an overall description of system architecture will be provided. Then we will focus on the Fact Extractor component, which exploits wrapper induction techniques to induce extraction rules on web pages semantically analyzed by other components. Special attention is on linguistic features.

2 Crossmarc Architecture

The overall CROSSMARC architecture [8] (see below Fig. 1) may be sketched as a pool of agents communicating via a dedicated XML language. Agents roles in the architecture are primarily related to three main tasks:

1. Process users' queries, perform user modeling, access the database and supply the user with product information

2. Extract information from the WEB: several processing steps are coordinated to find, retrieve, analyze and extract information from the Internet.

3. Store the extracted information in a database, in order to feed the system with the data to be later shown to the user

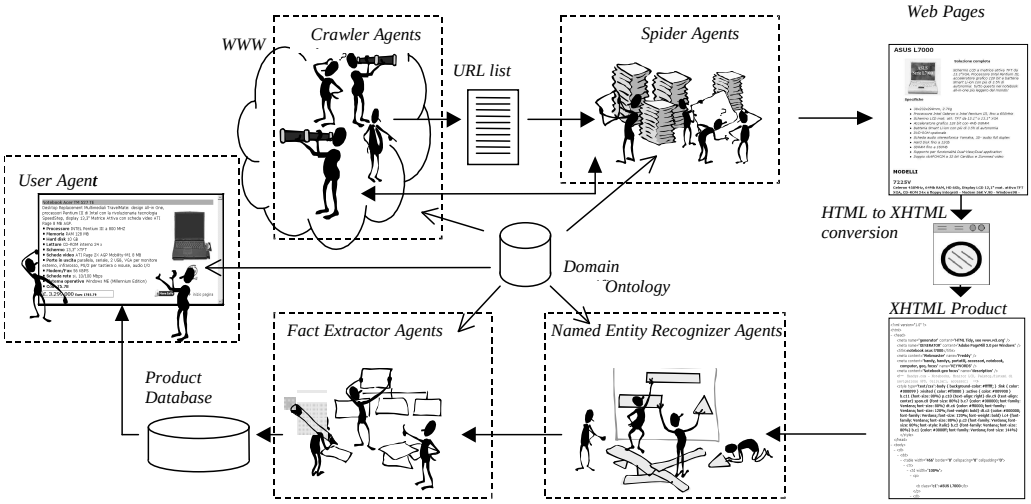

Fig. 1. Overall Crossmarc Architecture

Extraction agents can be divided into two broad categories, depending on their specific tasks:

- **Information retrieval agents (IR)**, which identify domain-relevant Web Sites (*focused crawling*) and return web pages inside these sites (*web spidering*) that are likely to contain the desired information;
- **Information Extraction** (**IE**) agents (one for each language) that process the retrieved web pages. There are specific roles for each step of the extraction process: a) *Named Entity Recognition and Classification* (NERC), i.e. recognition of concepts pertaining the domain of interest inside the web pages, b) identification of the number of products and their distribution in the web pages (*products demarcation*), c) *Fact Extraction* (FE), that is the extraction of products characteristics; all the information gathered during previous processing steps is merged in a XML data structure according to a common XML schema (the *Fact Extraction Schema*). Such a schema plays a pivotal role both in supporting interpretation of FE results by the product database feeder, and in providing consistency checking of the results.

Each agent commits to a shared ontology that drives its analysis throughout all the phases.
First of all, Named Entity Recognition and Classification (NERC) linguistic processors identify relevant entities in textual descriptions and categorize them according to the ontology [7], then inside the Fact Extraction and Normalization phase, these analyzed entities are aggregated to build a structured description of the identified product by exploiting the ontology organization. This description is composed of a set of features whose values are normalized to their canonical representation (as described into the ontology) for comparison purposes.
During the presentation of results to the end-user, correlations among different language lexicons and the ontology are exploited to adapt heterogeneous results to each language and locales.
The knowledge bases of the two domains (as well as lexicons for the four languages) [10], have been developed, accessed and maintained through a customized application based on Protégé-2000 API [5].

3 Wrapper Induction and Ontologies in the Crossmarc System

In the context of CROSSMARC, several FE components (one for language) have been developed by project partners. As a common characteristic, each Fact Extractor component implements wrapper induction techniques for extracting information pertaining to the products recognized inside each Web page. Boosted Wrapper Induction [4] has inspired the first version of the English Fact Extractor, STALKER [1] the Greek version of the Fact Extractor module, while the Italian one is a customized implementation of the Whisk algorithm [2].
In the following section more details on the Italian version of the Fact Extraction component and how it relies on semantic analysis carried on by other components of the CROSSMARC architecture will be provided.

3.1 CROSSMARC Italian Fact Extraction Component

WHISK [2] uses regular expressions as extraction patterns. It is not restricted to specific preprocessing of the text and hence it is good for structured, semi-structured and free texts. It induces a set of rules from hand-tagged training examples. WHISK rules are based on a form of regular expression patterns that identify both the context of relevant phrases and their delimiters for those phrases. Predefined domain-specific semantic classes are used, then applied to free text (the text is previously segmented into syntactic fields).

WHISK uses a covering algorithm inducing rules top-down, by first finding the most general one that covers the seed, then extending the rule by adding terms one at a time as long as it is below a certain threshold of error. Best performing rules are retained and the process is then reiterated until all the candidate extensions do not perform better than those produced in the previous step.
Although WHISK can learn either single or multi-slot rules, we considered the former ones: in fact target products may highly differ both in number and position of the features used for their description.

3.2 Boosting WHISK for CROSSMARC

The architecture of the Italian Fact Extractor (FE) component (see figure 2) consists of three different modules:

FE_Adapter: this module pre-processes the training set for the Learner and the Evaluator modules; it merges source web pages and annotation files into XML files, tokenizes the result (as WHISK works on tokenized input instances), and extracts from FE surrogate files a set of structured product descriptions, which will then be used by the training process to calculate the Laplacian Expected Error and to evaluate the output in terms of Precision and Recall statistics.

FE_Core: the FE core component. It performs both training and testing processes. These activities are implemented into a single module: in fact rules are continuously tested against the training set during the training phase, so training and testing need to be tightly coupled. If used in training mode, the output of this module is the set of induced rules; if used in testing mode, it produces a set of structured product descriptions, in the same format as the one produced by the FE_Adapter module (as they need to be compared during evaluation).

FE_Evaluator: it performs the evaluation of Whisk's product extractions on the test set by first identifying the number of correct extractions, then by computing and reporting statistics for precision and recall metrics.

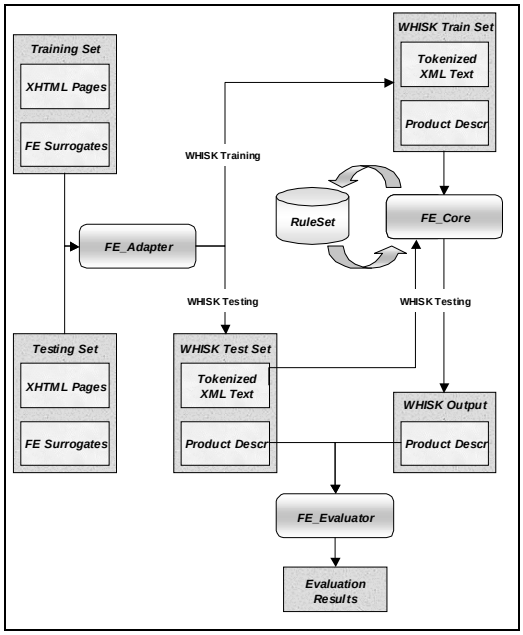

Fig 2: Interaction Diagram of Italian FE modules

The original algorithm has been customized to meet the specific needs of the CROSSMARC environment, through the following aspects:

a) Ontology Lookup. WHISK introduces the notion of Semantic Class to factorize disjunctive sets of terms into equivalence classes. A Semantic Class appears in an induced rule and provides a sort of generalization for the words it subsumes. In case a more complex analysis is needed to catch less evident phenomena, Whisk demands to other components (e.g., a syntactic parser) the task of providing such information in form of Semantic Tags wrapping recognized concepts.
These tags well fit on CROSSMARC needs, since all the Named Entities recognized by the NERC component are semantically classified into some ontological category (e.g. MANUFACTURER, PROCESSOR, CAPACITY, etc. in the Laptop Domain), while sets of Semantic Classes are defined on lists of lexical entries factorized according to the ontology. NERC components add to each XHTML page they process annotations for the named entities (NE), numeric expressions (NUMEX), time expressions (TIMEX) and terms they recognize. The type of NE, NUMEX, TIMEX is added as an attribute to the corresponding annotation. Here is an example of a tag inserted by a NERC component:

```
<NUMEX TYPE="LENGTH" onto-ref="OA-d0e1569" onto-attr="OA-d0e1569">
14.1 "
</NUMEX>
```

In the above examples 14.1” has been found and recognized as the length, expressed in inches, of something presented in the page. The FE component exploits the NERC annotations in order to identify which of the NEs, NUMEXs, TIMEXs, TERMs fill a specific fact slot inside a product description (e.g. which of the NUMEXs of type MONEY is the laptop’s PRICE etc…), according to the above mentioned XML FE Schema.
By considering NE categories as Semantic Classes allows the induction system to focus on relevant product characteristics, thus providing an higher level of interpretation together with a strong bias on the search space of the wrapper induction algorithm. As a result, inferred rules become more sensitive to semantic information with respect to instance specific delimiters, while enhancing their robustness towards heterogeneous data.

b) Limiting Search Space of Induction when adding terms. WHISK original algorithm was conceived to operate on fragments of the source material containing the information to be extracted, while all the FE components must operate on entire web pages. As the algorithm complexity increases with the number of considered features and with the dimension of instances, in [2] the search for terms to grow rules is limited to a window-size of *k* tokens around an extraction slot.
To find a good trade-off between accuracy of rules (the wider the windows, the larger is the space of induction), and the time needed to learn them, we adopted two different windows, depending on two defined parameters:

– *A Token window size (T_SIZE)*

– *A Semantic window size (S_SIZE)*

The following strategy has thus been implemented to allow for a more stable window for rule improvement:

1. A Token Window of *T_SIZE* size is created near the element to be extracted.

2. Tokens in the Token Window are converted to Semantic Elements (Semantic Classes or Tags)

S_SIZE elements (both Semantic classes and remaining tokens) are considered when adding terms to the WHISK rule expression

c) Laplacian Expected Error versus Precision: rule application strategy. The Laplacian Expected Error Rate (i.e *(e+1)/(n+1)* where *e* is the number of wrong extractions over *n* extractions), adopted by Soderland as performance metrics, has been preserved for evaluation of the partial rules created during rule expansion; it expresses a good trade-off between rule precision and recall. This measure has been used instead of the Precision to prevent Whisk from choosing a large set of very specific rules covering only very few cases, thus preferring a more lightweight and general-purpose ruleset. When dealing with system's test or online work a different criterion has been adopted, to consider actual precision of the rules and to prevent abuse of the less precise (but more general) ones. We thus applied the following strategy:

- Ruleset Construction

1. Each rule is characterized by its type (the kind of fact that it extracts), its Laplacian Expected Error and its Precision.
2. Rules from the ruleset are sorted by Precision
3. A threshold is set on Precision: induced rules with lower Precision are discarded; a different threshold for the Laplacian Expected Error is then considered: while it is not determinant for system accuracy, it helps to remove too specialized rules in order to enhance system performances.

- Rule appliance

For each rule, consider single extraction as a "candidate extraction"; for each candidate extraction:

1. discard the extraction if another candidate (whichever type it belongs) exists in the same span of tokens, else proceed to the next step.
2. discard the extraction if another candidate from a rule of the same type exists for the same product, else, proceed to the next step.
3. confirm the candidate as a valid extraction.

4 Evaluation of the Italian FE Component

The testing corpus for both NERC and FE components has been annotated by following a well known methodology ([3], [6]): a gold standard test set has been produced after comparison and merging annotations made by two different domain experts. The Italian test set for instance, consisted in a corpus of 100 web pages coming from 50 Italian sites of Laptop Vendors. A similar number of pages has been chosen for other languages.

Among the domain independent characteristics researched for the four languages in the 1st domain, product description category seems to affect the feasibility and difficulty of the IE tasks. Named Entity Recognition and Classification, for instance, is performed only within laptop product descriptions: a page including computer goods offers further to laptops, is more challenging than a page that includes only laptop descriptions. In fact, in the first case NERC component is more likely to recognize and classify non relevant names and expressions rather than in the second one, which consists only of descriptions of the actual products to be identified. Italian laptop vendor sites showed strong preference for one laptop description per page (45% of the corpus) and several laptop descriptions appearing in different lines/rows of a page (40%) with smaller preference for several laptop and other product descriptions appearing in different lines/rows of a page (8%). 42% of the pages in the Testing

corpus come from sites that do not appear in the Training corpus. A specific evaluation of each FE component in all the 4 different languages has been carried on: in table 1 evaluation results for the first domain (laptop computer offers) are summarized in precision and recall figures for all considered features for our Fact Extractor.

Table 1. Evaluation results for the Laptop Computers Domain on 4 different languages

FEATURE	ENGLISH		FRENCH		GREEK		ITALIAN	
	PREC	REC	PREC	REC	PREC	REC	PREC	REC
MANUFACTURER	0.89	1	0.99	1	1	1	1	0.99
PROCESSOR.	0.99	1	1	1	1	1	0.99	1
OPERATING SYSTEM	0.78	0.98	0.82	0.94	0.92	0.98	0.78	0.99
PROCESSOR SPEED	0.86	0.99	0.95	0.98	0.85	1	0.95	0.98
PRICE	0.99	1	1	1	1	1	1	1
HD CAPACITY	0.99	0.94	0.94	0.80	0.96	0.96	1	0.88
RAM CAPACITY	0.82	0.97	0.95	0.94	0.90	0.80	0.96	0.89
SCREEN SIZE	0.85	0.98	0.70	0.99	0.95	0.98	0.92	0.99
MODEL NAME	0.99	1	1	0.99	1	1	0.99	1
BATTERY TYPE	1	0.86	0.97	0.63	0.97	0.76	1	0.5
SCREEN TYPE	0.82	0.98	0.81	0.96	0.99	1	0.86	0.99
WEIGHT	0.98	1	0.96	1	1	1	0.92	1
AVERAGE VALUES	0.91	0,97	0.93	0.94	0.96	0.96	0.95	0.90

A straightforward comparison with Soderland's experiments on Whisk algorithm is not fully trustworthy, as we dealt with totally different domains and exploited richer linguistic analysis (and domain knowledge provided by the ontology). Bringing linguistic analysis in early processing phases provides more semantic evidences for the rule induction system reducing the dependency from specific page structures.
This results in rules with an higher level of coverage without significant loss in precision. All of these considerations motivated our architectural choices.
Moreover, the evaluation has been conducted inducing rules from a corpus of web pages with an high degree of heterogeneity to stress how the learned rules are less biased from rigid structure of the examined documents. Table 1 shows black-box evaluation of the FE components, as they are evaluated over error-free input from the previous processing steps (NERC and Product Demarcation), and does not provide sensitivity measures over noisy data. (Overall results for the IE system will be released by the end of the project foreseen for next Autumn).

5 Conclusions

At the cross point between knowledge-intensive IE systems with high maintenance needs and low-demanding machine learning algorithms, we have explored the possibility of combining the two approaches, leaving to the first one the knowledge of the domain required for the semantic analysis of the text while relying on the latter for explicit extraction of needed information.
Nowadays, in fact, the current trend in IE is in moving away from the rule-based approach, which relies on hand-crafted lexical resources and grammar rules, towards machine learning techniques in order to achieve swifter adaptation to new domains and text types.
Basing on this assumption, CROSSMARC reduces high system maintenance costs, which are closely related to modifications of tightly coupled rules and extraction methods, leaving only to Domain Experts and Knowledge Engineers the task of updating the ontology and lexicons of the domains, whilst machine learning techniques like Wrapper Induction can induce proper rules which exploit this

knowledge, thus furthering rapid adaptation to both new domains and changes in their conceptualisations.

References

[1] I. Muslea, S. Minton C. Knoblock "Hierarchical Wrapper Induction for Semistructured Sources". *Journal of Autonomous Agents and Multi-Agent Systems*. Vol. 4, pp. 93-114, 2001.

[2] Soderland S. "Learning Information Extraction Rules for semi-structured and free text. Machine learning. Volume 34 (1/3) pp. 233-272, 1999.

[3] Boisen, S., Crystal, M, Schwartz, R., Stone, R. and Wischedel, R. "Annotating Resources for Information Extraction" *LREC 2000* pp. 1211-1214

[4] Kushmerick N., "Finite-State Approaches to Web Information Extraction". in M.T. Pazienza Editor: *Information Extraction in the Web Era: Natural Language Communication for Knowledge Acquisition and Intelligent Information Agents*, LNAI2700, Springer-Verlag.

[5] N. F. Noy, R. W. Fergerson, & M. A. Musen. "The knowledge model of Protege-2000: Combining interoperability and flexibility". *2th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000)*, Juan-les-Pins, France, . 2000.

[6] MUC-7 (2001) http://www.itl.nist.gov/iad/894.02/related_projects/muc/

[7] C. Grover, S. McDonald, D. Nic Gearailt, V. Karkaletsis, D. Farmakiotou, G. Samaritakis, G. Petasis, M.T. Pazienza, M. Vindigni, F. Vichot and F. Wolinski (2002): "Multilingual XML-Based Named Entity Recognition for E-Retail Domains". *3rd International Conference on Language Resources and Evaluation (LREC 2002)*, Las Palmas, Spain

[8] M.T. Pazienza, M. Vindigni, (2002) "Mining linguistic information into an e-retail system" *Third International Conference on Data Mining Methods and Databases for Engineering, Finance and Other Fields*, Bologna, Italy

[9] M. T. Pazienza and M. Vindigni. "Language-based agent communication". *13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, Sigueza, Spain.

[10] M.T. Pazienza, A. Stellato, M. Vindigni, A. Valarakos, V. Karkaletsis (2003) "Ontology integration in a multilingual e-retail system" *HCI International 2003*, Crete, Greece.