On the Need to Bootstrap Ontology Learning with Extraction Grammar Learning

Georgios Paliouras

Institute of Informatics and Telecommunications, NCSR "Demokritos", Greece

Abstract. The main claim of this paper is that machine learning can help integrate the construction of ontologies and extraction grammars and lead us closer to the Semantic Web vision. The proposed approach is a bootstrapping process that combines ontology and grammar learning, in order to semi-automate the knowledge acquisition process. After providing a survey of the most relevant work towards this goal, recent research of the Software and Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos" in the areas of Web information integration, information extraction, grammar induction and ontology enrichment is presented. The paper concludes with a number of interesting issues that need to be addressed in order to realize the advocated bootstrapping process.

1 Introduction

The task of information extraction from text has been the subject of significant research in the past two decades. Primarily, this work has focussed on the extraction of very specific pieces of information from documents that belong in a very narrow thematic domain. The typical example is the extraction of information about mergers and acquisitions from business news articles, e.g. the information:

{Buying-company:"MacroHard Corp",
Company-bought:"Africa Off-Line Ltd",
Amount:"3 billion rupees"}

could be extracted from the text:

"MacroHard Corp bought Africa Off-Line Ltd for 3 billion rupees."

or from the text

"Africa Off-Line was sold to MacroHard. ... The acquisition has costed three Bil. Rup."

Based solely on this limited example, one can understand the difficulty of the information extraction task, which is arguably as hard as full text understanding. However, when limiting the domain and the information to be extracted there are various ways to avoid full understanding and produce good results with shallow parsing techniques. These techniques usually involve lexico-syntactic patterns,

coupled with a conceptual description of the domain and domain-specific lexicons. The manual construction and maintenance of these resources is a time-consuming process that can be partially automated with the use of learning techniques. For that reason, a significant part of the research in information extraction has refocussed on learning methods for the automatic acquisition of grammatical patterns, lexicons and even conceptual descriptions.

The rapid growth of the Web has brought significant pressure for practical information extraction solutions that promise to ease the problem of the user's overload with information. This has led to a new research direction, which aimed to take advantage of the uniform presentation style followed typically within particular Web sites. Information extraction systems designed for specific Web sites and based mainly on the HTML formatting information of Web pages have been termed wrappers. Despite the apparent ease of constructing wrappers, as opposed to free-text information extraction, the knowledge acquisition bottleneck remained, due to the frequent change in the presentation style of a specific Web site and most importantly due to the large number of different wrappers needed for any practical information integration system. This has led again to linguistically richer information extraction solutions and the use of learning methods.

More recently, a new goal was set for the information society: to move from the Web to the Semantic Web, which will contain many more resources than the Web and will attach machine-readable semantic information to all of these resources. The first steps towards that goal addressed the issue of knowledge representation for all this semantic information, which translated to the development of ontologies. Realizing the difficulty of designing the grand ontology for the world, research on the Semantic Web has focussed on the development of domain or task-specific ontologies which have started making their appearance in fairly large numbers. Having provided an ontology for a specific domain, the next step is to annotate semantically all related Web resources. If done manually, this process is very time-consuming and error-prone. Information extraction is the most promising solution for automating the annotation process. However, it comes along with the aforementioned knowledge acquisition bottleneck and the need for learning. At the same time, constructing and maintaining ontologies for various domains is also a hard knowledge acquisition task. In order to automate this process, the new research activity of ontology learning and population has emerged, which combines information extraction and learning methods.

Thus, information extraction makes use of various resources, among which a conceptual description of the domain, while at the same time ontology construction and maintenance is partly based on information extraction. The study of this interaction between the two processes and the role of learning in acquiring the required knowledge is the subject of this paper. The paper aims to initiate the interdisciplinary discussion and research that will lead to the uniform treatment of the problem of knowledge acquisition for information extraction and ontology maintenance. This effort is driven by the Semantic Web vision and the proposed vehicle is machine learning methods.

The rest of the paper is organized as follows: Section 2 highlights the state of the art in the related fields, focussing on key contributions that could facilitate the interdisciplinary effort. Section 3 presents related current research effort at the Software and Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos", where the author belongs[1]. Section 4 discusses key issues that should be addressed, in order to move this discussion forward. Finally section 5 summarizes the presented ideas.

2 State of the art

This section presents recent approaches in the areas of information extraction and ontology learning. Rather than providing an extensive survey of these areas, the focus is on those efforts that seem most promising in terms of the desired convergence of the different approaches.

2.1 Information Extraction

Practical information extraction systems focus on a particular domain and a narrow extraction task. Within this domain, they require a number of resources, in order to achieve the required analysis of the text and identify the pieces of information to be extracted. These resources mainly consist of grammars, lexicons and semantic models. The number and complexity of the resources that are used varies according to the approach. Early approaches focussed on linguistically rich resources, with the hope that they can capture a wide variety of linguistic phenomena (e.g. [22], [18]). This approach did not prove effective in practice, as the construction and the use of the resources was very expensive. As a result, a turn towards "lighter" task-specific approaches took place (e.g. [1], [2]). These approaches combined simple grammars, e.g. regular expressions, with existing generic dictionaries, e.g. the Wordnet [17], task-specific list names, known as gazetteers, and rather simple semantic models, template schemata. As a solution to the narrow scope of these approaches, the use of machine learning methods was proposed, which allowed for the quick customization of the resources to new extraction tasks (e.g. [53], [36], [44]). This approach was taken to the extreme with the introduction of Web site wrappers and the automatic learning of these (e.g. [25], [29]).

More recently, a move towards deeper analysis of the text has started, in order to improve the performance of the extraction systems, which seemed to have exhausted their capabilities, reaching what is known as the 60% performance barrier. These new efforts include the use of more complex grammars, e.g. HPSG, for deeper structural analysis, and the use of semantic models, e.g. domain-specific ontologies, for semantic disambiguation. These developments were made possible by the improvement in deep analysis methods and the increase in available computational power.

[1] SKEL: http://www.iit.demokritos.gr/skel

DFKI[2] research on information extraction [31] provides an interesting example of the progress towards deeper analysis. Starting with the use of finite-state transducers [32] for shallow analysis and moving onto the incorporation of domain-specific ontologies and Head-driven Phrase Structure Grammars (HPSG) for deep structural and semantic analysis [11]. The main aim of this work is to combine the best of shallow and deep analysis, i.e., speed and accuracy. In order to achieve that, various integration strategies have been studied, focussing the use of deep analysis to those cases that are most likely to help improve the accuracy of shallow analysis. This controlled use of deep analysis minimizes the computational overhead of the approach. Furthermore, initial efforts have been made [51] to use machine learning techniques to acquire basic semantic entities, such as domain-specific terms, and parts of the extraction grammars, in particular domain-specific lexico-syntactic patterns.

This multi-strategy approach that attempts to combine the advantages of shallow and deep analysis, as well as the strengths of automated knowledge acquisition through learning with the use of rich syntactic and semantic resources, is indicative of the general trend towards optimal combination of methods at various levels. Recent research at the University of Sheffield, UK also follows that trend, starting from simple wrapper-type information extraction approaches [8] and moving towards learning methods that incorporate more linguistic information, as well as domain-specific ontologies [9]. However, the focus of this work is still on the minimization of human effort in producing linguistic resources and as a result the extraction grammars are much simpler than the HPSGs.

A related strand of work has been using conceptual graphs as the representation for the extracted information. Typically, these approaches require deep syntactic analysis of the text and some domain knowledge, e.g. in the form of an ontology, in order to construct a graph for each sentence that is analyzed (e.g. [21], [45], [33]). These approaches face similar difficulties in acquiring the required knowledge for the mapping between syntax and semantics. Machine learning has been proposed as a partial solution to the problem, e.g. for learning the mapping rules between syntactic parses and conceptual graphs [54].

Most of the generic information extraction platforms have been extended to use ontologies, instead of the simpler template schemata of the past. Reeve and Han [40] provide a brief survey of the use of ontologies in information extraction platforms. For the majority of these systems though, ontologies are only used as a rich indexing structure, adding possibly to the variety of entities and relations that are sought for in the text (e.g. [41]). A notable exception is the concept of 'information extraction ontologies' [15], where the ontology plays the role also of a simple extraction system. This is achieved by incorporating lexical constraints in the ontology, using the data frame approach [14]. Data frames associate regular expressions with the domain-specific concepts, in order to allow for their direct extraction from the text. In a similar manner, Patrick [35] uses Systemic Functional Grammars, which combine high-level conceptual descriptions with

[2] DFKI: Deutsches Forschungszentrum fur Kuenstliche Intelligenz; http://www.dfki.de/

low-level textual and linguistic features. This work shows initial signs of convergence of extraction grammars with ontologies, whereby a conceptual structure incorporates sufficient information to be used for the extraction of instances of its concepts from text. Whether this extended structure is a grammar, an ontology or a completely different representation is of less importance.

2.2 Ontology learning

Machine learning methods have been used extensively to acquire various resources required for information extraction, in particular grammars and lexicons for part-of-speech tagging, sentence splitting, noun phrase chuncking, named-entity recognition, coreference resolution, sense disambiguation, etc. They have also been used to acquire the lexico-syntactic patterns or grammars that are used for information extraction (e.g. [53], [36], [44]), in particular the simpler regular expressions used in wrappers (e.g. [25], [29]). More recently this line of work has been extended to target the semantic model needed for information extraction, which is in most cases an ontology. Thus, the new research activity of ontology learning and population has emerged.

Ontology learning methods vary significantly according to their dependence on linguistic processing of the training data. At one extreme, the learning process is driven completely by the results of language processing [5]. Following this approach, the OntoLT toolkit allows the user to define or tune linguistic rules that are triggered by the data and result in the modification of the ontology. In other words, each rule defines linguistic preconditions, which, if satisfied by a sentence, lead to a change in the existing ontology, usually extending the ontology with new concepts and their properties. This is a deductive approach to concept learning, which has been the subject of dispute in the early days of machine learning, as it has been argued that it does not cause generalization and therefore is not learning at the knowledge level [13]. Nevertheless, it can be an effective method for enriching an ontology, although it requires significant expertise in defining the linguistic rules. The Ontology Express system [34] follows a similar linguistic approach to ontology learning, with two notable exceptions: (a) it concentrates on the discovery of new concepts, based on the identification of specific patterns in the text that usually denote particular relations, e.g. introduction of a new term as a subtype of an existing one, (b) it uses a frequency-based filter and non-monotonic reasoning to select the new concepts and add them to the ontology.

Term identification and taxonomic association of the discovered terms has been the most researched aspect of ontology learning. One of the earliest systems to adopt this approach was ASIUM [16], which uses lexico-syntactic patterns, in the form of subcategorization frames that are often used for information extraction, in order to identify interesting entities in text. Starting with a few generic patterns that are instantiated in the text, ASIUM uses syntactic-similarity in order to cluster terms into groups of similar functionality in the text. This process is repeated, building the taxonomy of an ontology in a bottom-up fashion. A similar approach is followed by the DOGMA system [42]. Verb-driven syntactic

relations, similar to generic subcategorization frames, are used to cluster terms with syntactic similarity. Term clustering is also employed by the OntoLearn system [30]. However, OntoLearn differs from the other two systems in two ways: (a) it combines statistics about term occurrence with linguistic information for the identification of terms, and most importantly (b) clustering is based on semantic interpretation through a mapping of the terms onto an existing ontology, such as Wordnet. Thus, the resulting ontology is a domain-specific subset of the generic one. Wordnet is also used in [51] to identify an initial set of examples of the hyponymy relation in an untagged corpus. Given these examples, generic extraction patterns are learned. These patterns are combined with the results of a statistical term identification method and the collocation patterns learned by a different statistical method, to provide a set of candidate concepts for the new ontology. More recently, Wordnet and lexico-syntactic patterns have been combined in [7] using a simple voting strategy, in order to identify terms and organize them in a taxonomy. Despite the simplicity of the voting strategy, the combination of various evidence from different methods seems to provide added value to the ontology learning process.

Another promising approach to ontology learning is based on the use of Formal Concept Analysis for term clustering and concept identification. In [43] concept lattices are constructed from data with the use of a knowledge acquisition method known as 'ripple-down rules'. The acquired conceptual structures are then used to define domain ontologies, with the cooperation of a human expert. In a related approach, Corbett [10] represents ontologies with the use of Conceptual Graphs and uses Conceptual Graph Theory, in order to automate ontology learning through merging of conceptual graphs. Given the use of conceptual graphs in information extraction from text, as discussed in 2.1, this approach provides an interesting link between extraction and ontology learning. For instance, a clustering approach for conceptual graphs, such as the one presented in [52], could be used to learn ontologies, in the form of contextual graphs, from text.

The highlights of ontology learning research presented in this subsection indicate the close relation between extraction patterns and concept discovery. One usually learns the extraction patterns at the same time as identifying new terms and relations among them with the aim to construct or refine an ontology. The work of Hahn and Markó [19] emphasizes this interaction, providing a method to learn grammatical in parallel with conceptual knowledge. Adopting a deductive learning approach, like OntoLT, the proposed method refines a lexicalized dependency grammar and a KL-ONE-type conceptual model, through the analysis of text and the qualitative assessment of the results.

The interaction between information extraction and ontology learning has also been modelled at a methodological level as a bootstrapping process that aims to improve both the conceptual model and the extraction system through iterative refinement. In [27] the bootstrapping process starts with an information extraction system that uses a domain ontology. The system is used to extract information from text. This information is examined by an expert, who may

decide to modify the ontology accordingly. The new ontology is used for further information extraction and ontology enrichment. Machine learning assists the expert by suggesting potentially interesting taxonomic and non-taxonomic relations between concepts. Brewster et al. [3] propose a slightly different approach to the bootstrapping process. Starting with a seed ontology, usually small, a number of concept instances are identified in the text. An expert separates these as examples and counter-examples which are then used to learn extraction patterns. These patterns are used to extract new concept instances and the expert is asked to re-assess these. When no new instances can be identified, the expert examines the extracted information and may decide to update the ontology and restart the process. The main difference between the two approaches is in the type of extraction system that is used, which is linguistically richer in the case of [27] and uses the ontology as a component.

3 Recent research results by SKEL

The Software and Knowledge Engineering Laboratory (SKEL) of the Institute of Informatics and Telecommunications in the National Center for Scientific Research "Demokritos" has set as its main goal for the past decade to advance knowledge technologies that are required for overcoming the obstacle of information overload on the Web. Towards that goal, it has produced innovative research results in the whole chain of technologies employed by intelligent information integration systems: information gathering (retrieval, Web crawling), information extraction (named entity recognition and classification, role identification, wrappers), personalization (user communities and stereotypes). The recent emphasis of our research has been on the automation of intelligent system development, customization and maintenance, which involves mainly the employment of machine learning methods for knowledge acquisition.

This section highlights SKEL's most recent research activity in the area of machine learning for information extraction and ontology enrichment. It starts by presenting briefly the CROSSMARC architecture for information integration, which is the main result of the European research project CROSSMARC and provides the framework for our research in this area. It then moves on to present briefly our meta-learning approach to information extraction from Web pages, an efficient learning method for context-free grammars and a bootstrapping methodology for ontology enrichment.

3.1 The CROSSMARC approach to Web information integration

CROSSMARC (Cross-lingual Multi Agent Retail Comparison)[3] was a European research project that was completed at the end of 2003. The main result of CROSSMARC was an open, agent-based architecture for cross-lingual information integration, incorporating the full chain of technologies involved in the

[3] http://www.iit.demokritos.gr/skel/crossmarc/

process. Initially, CROSSMARC was meant to focus on retail comparison systems that collect product information from various suppliers and present it to customers in a localized and personalized manner. In addition to this application however, the CROSSMARC architecture has proven equally useful for other information integration tasks, such as employment search engines. Figure 1 presents the CROSSMARC architecture.

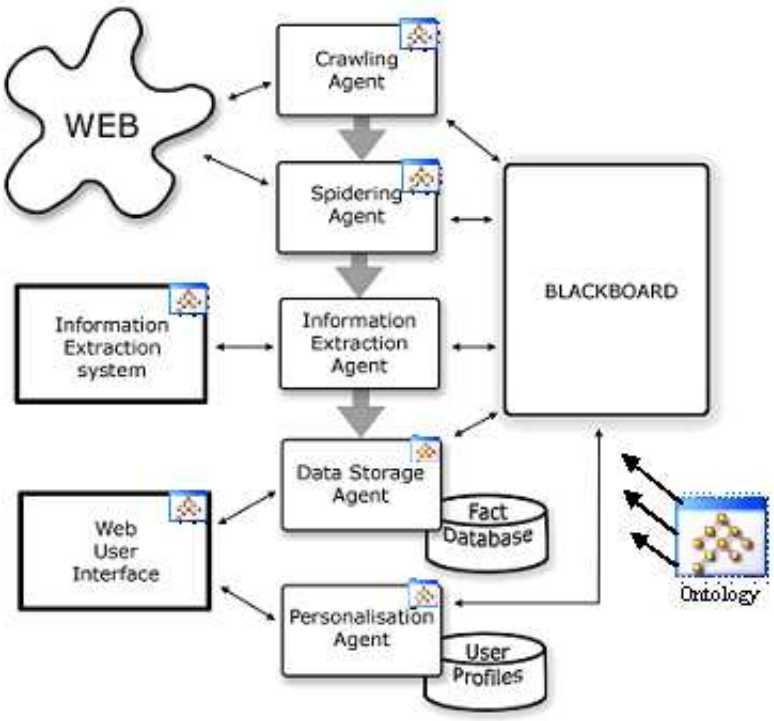

Fig. 1. CROSSMARC's agent based architecture.

As mentioned above, CROSSMARC implements the full information integration process, using independent agents that communicate via a blackboard and share the same domain ontology. The information gathering stage is separated into a crawling and a spidering step, collecting interesting sites from the Web and relevant pages from these sites, respectively. Machine learning is used to learn to identify relevant pages and the most promising paths to those pages. The information extraction agent serves as a controller for a number of different information extraction systems, each handling a different language (English, French, Italian and Greek are currently covered). The results of information extraction are stored into the fact database, which is accessed by the

end-users through a personalized Web interface. The agent-based design of the CROSSMARC architecture allows it to be open, distributed and customizable. The agents implementing each step of the process can be replaced by any other tool with the same functionality that respects the XML-based communication with the blackboard and the ontology. Furthermore, new information extraction systems, covering different languages can easily be connected to the information extraction agent. The ontology also plays an essential role in all stages, providing terms for modeling the relevance of Web pages, language-independent fact extraction, parameterization of the user models, etc. By collecting domain-specific knowledge in the ontology, the various agents become less dependent on the domain. More details about the CROSSMARC architecture and the prototype can be found in [24] and [47].

3.2 Meta-learning for Web information extraction

As we have seen in section 2 several approaches to information extraction and ontology learning attempt to combine the strengths of multiple methods in order to obtain better performance. Following this basic idea, our research in the area of Web information extraction has focussed on the combination of different learning methods in a meta-learning framework, aiming to improve recognition performance. For this reason, we have developed a stacked generalization framework that is suitable for information extraction, rather than classification which is the typical use of this approach. Figure 2 illustrates the use of the stacking framework proposed in [46], both at training and at run-time.

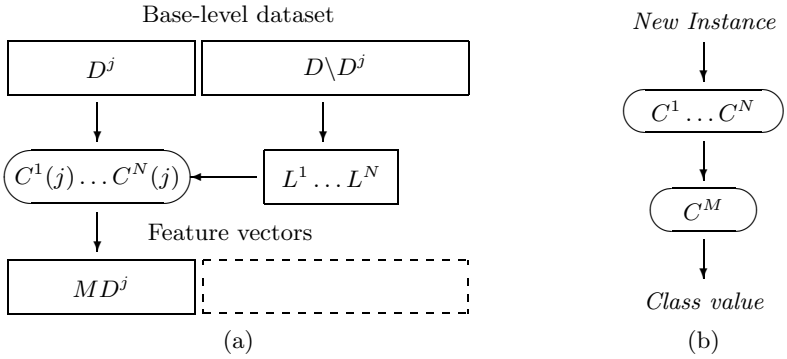

Fig. 2. (a) Illustration of the J-fold cross-validation process for creating the meta-level dataset. (b) The stacking framework at runtime.

At training time, the usual cross-validation approach of stacked generalization is followed, which trains all base-level learners ($L^1 \ldots L^N$) on various subsets of the training dataset ($D \backslash D^j$) and applies the learned systems ($C^1(j) \ldots C^N(j)$) on the unseen parts of the dataset (D^j), in order to construct the meta-level

dataset (MD^j). However, in the case of information extraction the trained systems may extract different or contradictory information from the same subset of the data. Based on the confidence scores of the information extraction systems, the proposed framework combines their results into a common dataset that is suitable for training a classifier to choose whether to accept or reject an extracted piece of information and if accepted to recognize its type. This approach has led to considerable improvement in recognition performance, which is due to the complementarity of the trained base-level systems.

3.3 Grammar induction

The importance of grammars for information extraction has become apparent in the description of relevant systems in section 2. With the exception of simple regular patterns, the acquisition of grammars using learning methods is limited to the refinement of specific parameters of hand-made grammars. This is due to the fact that the learning of more complex grammars from text is a hard task. Even harder is the induction of these grammars from positive only examples, which is practically the only type of example that a human annotator can provide. This is the reason why there are very few learning methods that deal with this problem, which are usually only applicable to datasets of small size and complexity. In an attempt to overcome this problem we have developed the e-GRIDS algorithm [37], which is based on the same principles as the GRIDS [26] and the SNPR [50] algorithms, but improves them substantially, in order to become applicable to realistic problems.

e-GRIDS performs a beam search in the space of grammars that cover the positive examples, guided by the MDL principle. In other words, it favors simpler grammars, in the sense that the sum of their code length and the code length of the data, assuming knowledge of these grammars, should be small. The starting state for the search is the most specific grammar, which covers only the training data. The search operators compress and generalize the grammars, by merging symbols and creating new ones. The latest version of e-GRIDS, called eg-GRIDS [38], replaces the beam search with a genetic one, within which the grammar-modification operators are treated as mutation operators. Figure 3 depicts graphically the eg-GRIDS architecture. The use of genetic search has provided a speed-up of an order of magnitude, facilitating the inclusion of more operators that allow the algorithm to search a larger part of the space and produce much better results. Thus, eg-GRIDS can handle larger datasets, and produce better estimates of the "optimal grammar".

3.4 Ontology enrichment

Our approach to ontology enrichment [48] follows the basic bootstrapping methodology presented in section 2. Figure 4 illustrates the proposed methodology. The bootstrapping process commences with an existing domain ontology, which is used to annotate a corpus of raw documents. In this manner, a training corpus for information extraction is formed, without the need for a human annotator.

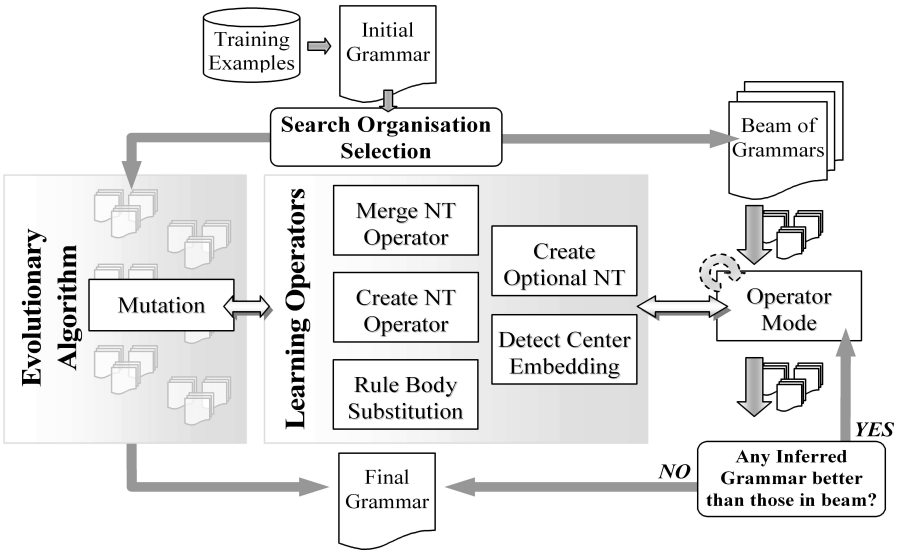

Fig. 3. The architecture of the eg-GRIDS algorithm. (NT: Non-Terminal symbol)

This can lead to a significant speed-up in the development of the information extraction system. The trained system usually generalizes beyond the annotated examples. Therefore, if applied again on the corpus it provides some new instances that do not appear in the initial ontology. These instances are screened by an expert who is responsible for maintaining the ontology. Once the ontology is updated, it can be used again to annotate a new training corpus that will lead to a new information extraction system. This process is repeated until no new instances are added to the ontology. Our initial experiments have shown that this approach works impressively well, even when the initial ontology is very sparsely populated.

As a further improvement of this method, COCLU [49], a novel compression-based clustering algorithm, was developed, which is responsible for identifying lexical variations of existing instances and clustering the lexical variations of new instances. This improvement minimizes the human effort in ontology enrichment, as the extracted instances can be treated in groups of lexical synonyms.

4 Discussion

The combination of ontology learning and information extraction under a bootstrapping framework of iterative refinement seems to be a promising path towards the Semantic Web vision. The use of machine learning for the (partial) automation of knowledge acquisition also seems to be a vital part of this process. However, there are various issues that arise under this framework and need to be researched, in order to arrive at theoretically sound and practically effective

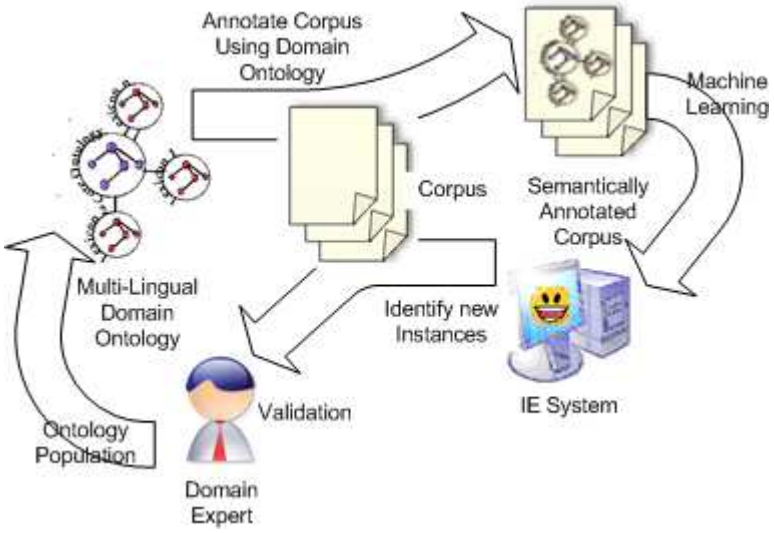

Fig. 4. Ontology enrichment methodology. (IE: Information Extraction)

solutions. This section raises some of these issues, together with some initial thoughts about them.

4.1 Knowledge representation issues

The most straightforward option in terms of knowledge representation is to keep the extraction grammars and the ontologies as separate entities, which will allow us to take full advantage of the work that has been done in each of the two research areas. This is the approach that is adopted in most of the work following the bootstrapping paradigm. However, in section 2 we have also seen some work on alternative representations that combine conceptual and syntactic knowledge under the same representation. This option would simplify the bootstrapping process, but may also have disadvantages, such as the fact that the combined representation needs to be task-specific, which limits the ability of the ontology to provide interoperability and knowledge sharing. Therefore, the combination of the ontology with the grammar remains an open issue to be studied.

If a combination is the preferred solution, a number of new questions arise, such as what type of grammars and what type of ontology one should use. A number of solutions already exist, as we have seen in section 2. However, the choice of an appropriate solution depends largely on the extraction task and the need for syntactic and conceptual support. Recently, the use of more complex grammars and ontologies has been proposed as a solution to the barrier in the performance of extraction systems. Nevertheless, there is still a number of problems that may be addressed with simpler solutions. Therefore, work on the typology of the extraction tasks and the need for resources is necessary.

Finally, in those cases where a combined representation is preferred, we should also study solutions that are inspired by the early work on knowledge representation, e.g. frames and semantic nets. Formal concept analysis and conceptual graphs, as well as probabilistic graphical models are examples of such representations, which have advanced considerably since their conception. Such solutions have started being studied in the context of the bootstrapping framework [6] and may prove very effective in practice.

4.2 Machine learning issues

The choice of knowledge representation affects directly the machine learning methods that will be used for knowledge acquisition. If grammars and ontologies are kept separate, the main question is which aspects of the ontology and the grammar will be learned and which will be provided by a human. So far, we have seen almost full automation for simple grammatical patterns and basic conceptual entities and relations. However, there is a host of other methods, such as those that induce context-free grammars, which have not been studied sufficiently in the context of information extraction. An additional issue is whether grammar learning can assist ontology learning and vice versa, i.e., can the elements of the representation acquired through learning be useful at both the conceptual and syntactic level? The answer to these questions depends very much on the type of training data that is available.

Supervised learning of complex representations requires data that may not be possible to acquire manually. Therefore, efforts to automate the generation of training data, such as in [48], are very interesting. Furthermore, unsupervised or partially supervised methods may prove particularly useful. Along these lines, we also need to find better ways to take into account existing background knowledge. Deductive learning methods are the extreme solution in that sense, but inductive learning methods can also benefit from existing knowledge resources.

If we opt for combined representations, it is more than likely that the learning methods will need to be extended, or existing methods will need to be combined in an intelligent way. Multi-strategy learning can prove particularly useful in this respect, as it aims to combine the strengths and special features of different learning methods in the best possible way.

4.3 Content type issues

Another major issue that affects directly the typology of extraction and learning tasks is what type of content we want to process. We have already seen that the semi-structured format of Web data can facilitate significantly the information extraction task and the learning of Web site wrappers. Further to that, some semantically annotated content has started appearing. We need to think about how we can make use of that, e.g. as training data for learning new extraction systems, or as background knowledge. An interesting alternative presented in [12] is to treat a set of resources, linked through RDF annotations, as a graph and construct a conceptual model from it.

Multimedia content is increasing on the Web. We need to examine more carefully the task of extracting information from such data, which is more demanding and less studied than text. Can we make assumptions that will allow us to produce practical extraction systems for multimedia data? At what conceptual level can we expect the extracted information to be placed? We need to go beyond the basic low-level features, but how feasible is object recognition within specific domains? Can multimedia ontologies assist in that process (e.g. [20]). There is even some initial work on enriching multimedia ontologies, through the processing of multimedia data [28]. Extraction grammars and the bootstrapping process advocated in this paper could be particularly useful in that respect. Initial ideas on how this can be achieved are presented in [23].

5 Summary

This paper advocates the need for a bootstrapping process, combining ontology learning and grammar learning, in order to semi-automate the construction of ontologies and information extraction systems. The aim of the paper was to present the most relevant work for this purpose, focussing on recent work at SKEL, the laboratory where the author belongs. Several related strands of SKEL research were presented: Web information integration, meta-learning for Web information extraction, induction of context-free grammars and ontology enrichment, through bootstrapping with information extraction learning. Finally, several research issues that need to be addressed towards the realization of the bootstrapping process were discussed. In summary, the main claim of this paper is that machine learning is the vehicle that could help integrate the construction of ontologies and extraction grammars and lead us closer to the Semantic Web.

Acknowledgments

This paper includes ideas and work that are not solely of the author. A number of current and past SKEL members have been involved in the presented work. This research was partially funded by the EC through the IST-FP5 project CROSSMARC (Cross-lingual Multi Agent Retail Comparison), contract number IST-2000-25366. The development of the CROSSMARC architecture has been carried out jointly by the partners of the project: National Center for Scientific Research "Demokritos" (GR), VeltiNet A.E. (GR), University of Edinburgh (UK), Universita di Roma Tor Vergata (IT), Lingway (FR).

References

1. Douglas E. Appelt, Jerry R. Hobbs, John Bear, David J. Israel, and Mabry Tyson. Fastus: A finite-state processor for information extraction from real-world text. In Ruzena Bajcsy, editor, *IJCAI*, pages 1172–1178, 1993.
2. Daniel M. Bikel, Scott Miller, Richard L. Schwartz, and Ralph M. Weischedel. Nymble: a high-performance learning name-finder. In *ANLP*, pages 194–201, 1997.

3. Christopher Brewster, Fabio Ciravegna, and Yorick Wilks. User-centred ontology learning for knowledge management. In Birger Andersson, Maria Bergholtz, and Paul Johannesson, editors, *NLDB*, volume 2553 of *Lecture Notes in Computer Science*, pages 203–207. Springer, 2002.
4. Paul Buitelaar, Siegfried Handschuh, and Bernardo Magnini, editors. *Proceedings of the ECAI Ontology Learning and Population Workshop, Valencia, Spain, 22-24 August.*, 2004.
5. Paul Buitelaar, Daniel Olejnik, and Michael Sintek. A protégé plug-in for ontology extraction from text based on linguistic analysis. In Christoph Bussler, John Davies, Dieter Fensel, and Rudi Studer, editors, *ESWS*, volume 3053 of *Lecture Notes in Computer Science*, pages 31–44. Springer, 2004.
6. Philipp Cimiano, Andreas Hotho, Gerd Stumme, and Julien Tane. Conceptual knowledge processing with formal concept analysis and ontologies. In Peter W. Eklund, editor, *ICFCA*, volume 2961 of *Lecture Notes in Computer Science*, pages 189–207. Springer, 2004.
7. Philipp Cimiano, Lars Schmidt-Thieme, Aleksander Pivk, and Steffen Staab. Learning taxonomic relations from heterogeneous evidence. In Buitelaar et al. [4].
8. Fabio Ciravegna. Adaptive information extraction from text by rule induction and generalisation. In Bernhard Nebel, editor, *IJCAI*, pages 1251–1256. Morgan Kaufmann, 2001.
9. Fabio Ciravegna, Alexiei Dingli, David Guthrie, and Yorick Wilks. Integrating information to bootstrap information extraction from web sites. In Subbarao Kambhampati and Craig A. Knoblock, editors, *IIWeb*, pages 9–14, 2003.
10. Dan Corbett. Interoperability of ontologies using conceptual graph theory. In *ICCS*, volume 3127 of *Lecture Notes in Computer Science*, pages 375–387. Springer, 2004.
11. Berthold Crysmann, Anette Frank, Bernd Kiefer, Stefan Mueller, Günter Neumann, Jakub Piskorski, Ulrich Schäfer, Melanie Siegel, Hans Uszkoreit, Feiyu Xu, Markus Becker, and Hans-Ulrich Krieger. An integrated archictecture for shallow and deep processing. In *ACL*, pages 441–448, 2002.
12. Alexandre Delteil, Catherine Faron, and Rose Dieng. Building concept lattices by learning concepts from rdf graphs annotating web documents. In Priss et al. [39], pages 191–204.
13. T. G. Dietterich. Learning at the Knowledge Level. *Machine Learning*, 1(3):287–316, 1986.
14. David W. Embley. Programming with data frames for everyday data items. In *NCC*, page 301305, 1980.
15. David W. Embley. Towards semantic understanding – an approach based on information extraction ontologies. In Klaus-Dieter Schewe and Hugh E. Williams, editors, *ADC*, volume 27 of *CRPIT*, page 3. Australian Computer Society, 2004.
16. David Faure and Claire Nedellec. Knowledge acquisition of predicate argument structures from technical texts using machine learning: The system asium. In Dieter Fensel and Rudi Studer, editors, *EKAW*, volume 1621 of *Lecture Notes in Computer Science*, pages 329–334. Springer, 1999.
17. Christiane Fellbaum, editor. *WordNet An Electronic Lexical Database*. Bradford Books, 1998.
18. R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham, and Y. Wilks. University of sheffield: Description of the lasie system as used for muc-6. In *MUC-6*, pages 207–220, 1995.

19. Udo Hahn and Kornél G. Markó. An integrated, dual learner for grammars and ontologies. *Data Knowl. Eng.*, 42(3):273–291, 2002.
20. Asaad Hakeem, Yaser Sheikh, and Mubarak Shah. Casee: A hierarchical event representation for the analysis of videos. In Deborah L. McGuinness and George Ferguson, editors, *AAAI*, pages 263–268. AAAI Press / The MIT Press, 2004.
21. Jeff Hess and Walling R. Cyre. A cg-based behavior extraction system. In William M. Tepfenhart and Walling R. Cyre, editors, *ICCS*, volume 1640 of *Lecture Notes in Computer Science*, pages 127–139. Springer, 1999.
22. Paul S. Jacobs and Lisa F. Rau. Scisor: Extracting information from on-line news. *Communications of the ACM*, 33(11):88–97, 1990.
23. Vangelis Karkaletsis, Georgios Paliouras, and Constantine D. Spyropoulos. A bootstrapping approach to knowledge acquisition from multimedia content with ontology evolution. In Timo Honkela and Olli Simula, editors, *AKRR*. Helsinki University of Technology, 2005.
24. Vangelis Karkaletsis and Constantine D. Spyropoulos. Cross-lingual information management from web pages. In *PCI*, 2003.
25. Nicholas Kushmerick. Wrapper induction: Efficiency and expressiveness. *Artificial Intelligence*, 118(1-2):15–68, 2000.
26. Pat Langley and Sean Stromsten. Learning context-free grammars with a simplicity bias. In Ramon López de Mántaras and Enric Plaza, editors, *ECML*, volume 1810 of *Lecture Notes in Computer Science*, pages 220–228. Springer, 2000.
27. Alexander Maedche and Steffen Staab. Mining ontologies from text. In Rose Dieng and Olivier Corby, editors, *EKAW*, volume 1937 of *Lecture Notes in Computer Science*, pages 189–202. Springer, 2000.
28. Joseph Modayil and Benjamin Kuipers. Bootstrap learning for object discovery. In *IROS*. IEEE Press, 2004.
29. Ion Muslea, Steven Minton, and Craig A. Knoblock. A hierarchical approach to wrapper induction. In *Agents*, pages 190–197, 1999.
30. Roberto Navigli and Paola Velardi. Learning domain ontologies from document warehouses and dedicated websites. *Computational Linguistics*, 30(2), 2004.
31. Guenter Neumann and Feiyu Xu. Course on intelligent information extraction. In *ESSLI*, 2004.
32. Günter Neumann and Jakub Piskorski. A shallow text processing core engine. *Computational Intelligence*, 18(3):451–476, 2002.
33. Stéphane Nicolas, Bernard Moulin, and Guy W. Mineau. Sesei: A cg-based filter for internet search engines. In Aldo de Moor, Wilfried Lex, and Bernhard Ganter, editors, *ICCS*, volume 2746 of *Lecture Notes in Computer Science*, pages 362–377. Springer, 2003.
34. Norihiro Ogata and Nigel Collier. Ontology express: Statistical and non-monotonic learning of domain ontologies from text. In Buitelaar et al. [4], pages 19–24.
35. Jon Patrick. The scamseek project: Text mining for finanical scams on the internet. In S.J. Simoff and G.J. Williams, editors, *ADMC*, pages 33–38, 2004.
36. Georgios Petasis, Alessandro Cucchiarelli, Paola Velardi, Georgios Paliouras, Vangelis Karkaletsis, and Constantine D. Spyropoulos. Automatic adaptation of proper noun dictionaries through cooperation of machine learning and probabilistic methods. In Nicholas J. Belkin, Peter Ingwersen, and Mun-Kew Leong, editors, *SIGIR*, pages 128–135. ACM, 2000.
37. Georgios Petasis, Georgios Paliouras, Vangelis Karkaletsis, Constantine Halatsis, and Constantine D. Spyropoulos. e-grids: Computationally efficient grammatical inference from positive examples. *Grammars*, 2004.

38. Georgios Petasis, Georgios Paliouras, Constantine D. Spyropoulos, and Constantine Halatsis. eg-grids: Context-free grammatical inference from positive examples using genetic search. In Georgios Paliouras and Yasubumi Sakakibara, editors, *ICGI*, volume 3264 of *Lecture Notes in Computer Science*, pages 223–234. Springer, 2004.
39. Uta Priss, Dan Corbett, and Galia Angelova, editors. *Conceptual Structures: Integration and Interfaces, 10th International Conference on Conceptual Structures, ICCS 2002, Borovets, Bulgaria, July 15-19, 2002, Proceedings*, volume 2393 of *Lecture Notes in Computer Science*. Springer, 2002.
40. Lawrence Reeve and Hyoil Han. The survey of semantic annotation platforms. In *ACM/SAC*, 2005.
41. D. Reidsma, J. Kuper, T. Declerck, H. Saggion, and H. Cunningham. Cross document ontology based information extraction for multimedia retrieval. In *Supplementary proceedings of the ICCS03*, Dresden, 2003.
42. Marie-Laure Reinberger and Peter Spyns. Discovering knowledge in texts for the learning of dogma-inspired ontologies. In Buitelaar et al. [4], pages 19–24.
43. Debbie Richards. Addressing the ontology acquisition bottleneck through reverse ontological engineering. *Knowledge and Information Systems*, 6(4):402–427, 2004.
44. Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *AAAI/IAAI*, pages 474–479, 1999.
45. G. Angelova S. Boytcheva, P. Dobrev. Cgextract: Towards extraction of conceptual graphs from controlled english. In *Supplementary proceedings of the ICCS01*, Stanford, USA, 2001.
46. Georgios Sigletos, Georgios Paliouras, Constantine D. Spyropoulos, and Takis Stamapoulos. Stacked generalization for information extraction. In Ramon López de Mántaras and Lorenza Saitta, editors, *ECAI*, pages 549–553. IOS Press, 2004.
47. Constantine D. Spyropoulos, Vangelis Karkaletsis, Claire Grover, Maria-Teresa Pazienza, Dimitris Souflis, and Jose Coch. Final report of the project crossmarc (cross-lingual multi agent retail comparison). Technical report, 2003.
48. Alexandros G. Valarakos, Georgios Paliouras, Vangelis Karkaletsis, and George A. Vouros. Enhancing ontological knowledge through ontology population and enrichment. In Enrico Motta, Nigel Shadbolt, Arthur Stutt, and Nicholas Gibbins, editors, *EKAW*, volume 3257 of *Lecture Notes in Computer Science*, pages 144–156. Springer, 2004.
49. Alexandros G. Valarakos, Georgios Paliouras, Vangelis Karkaletsis, and George A. Vouros. A name-matching algorithm for supporting ontology enrichment. In George A. Vouros and Themis Panayiotopoulos, editors, *SETN*, volume 3025 of *Lecture Notes in Computer Science*, pages 381–389. Springer, 2004.
50. Gerry Wolff. Grammar discovery as data compression. In *AISB/GI*, pages 375–379, 1978.
51. Feiyu Xu, Daniela Kurz, Jakub Piskorski, and Sven Schmeier. Term extraction and mining of term relations from unrestricted texts in the financial domain. In *BIS*, 2002.
52. Manuel Montes y Gómez, Alexander F. Gelbukh, and Aurelio López-López. Text mining at detail level using conceptual graphs. In Priss et al. [39], pages 122–136.
53. Roman Yangarber, Winston Lin, and Ralph Grishman. Unsupervised learning of generalized names. In *COLING*, 2002.
54. Lei Zhang and Yong Yu. Learning to generate cgs from domain specific sentences. In Harry S. Delugach and Gerd Stumme, editors, *ICCS*, volume 2120 of *Lecture Notes in Computer Science*, pages 44–57. Springer, 2001.