

# Methods for Ontology-Driven Integration

Perakath Benjamin, Richard Mayer, Peng Xu,  
Belita Gopal, Dan Corlette, and Olga Bagatourova  
Knowledge Based Systems, Inc.,  
College Station, TX 77840, U.S.A.

*Abstract—This paper describes the motivations, approach, and architecture for using ontologies in knowledge extraction and in applications that assist situated agents in complex information integration tasks. Our approach applies ontologies along with semantic analysis methods to extract task relevant knowledge from distributed, unstructured text sources. This knowledge is then applied to assist in information integration, sharing, and situation-awareness applications. We have developed a software architecture that provides support for automated ontology extraction, ontology conflict analysis and mapping, and knowledge integration and sharing. We are currently designing and building multiple applications that validate the practical benefits of this research.*

**Keywords:** knowledge, ontology matching, text mining, ontology extraction.

## 1.0 Introduction

This paper describes the motivations, approach, and architecture for using ontologies in knowledge extraction and in applications that assist situated agents in complex information integration tasks. Our approach applies ontologies for knowledge extraction along with semantic analysis methods to extract task relevant knowledge from distributed, unstructured text sources. This knowledge is then applied to assist the agent in information integration, sharing, and situation-awareness applications.

Innovative aspects of this research include (i) the application of ontology-assisted text mining methods for knowledge extraction from unstructured text sources; (ii) ontology conflict analysis and mapping methods to facilitate semantic information integration and information sharing; and (iii) text analysis mechanisms that will enable the automatic revisions to

the extracted knowledge in uncertain and dynamic task environments.

This paper is organized as follows. Section 2 addresses the motivations and challenges of complex situation assimilation and information integration research. Section 3 details our solutions for ontology extraction, knowledge integration, and data mediation. Section 4 provides an overview of the software architecture that integrates the components described in Section 3. Finally, Section 5 concludes the paper and discusses future research directions.

## 2.0 Motivations

There are three roadblocks in complex situation assimilation and information integration: semantic inaccessibility, logical disconnectedness, and consistency maintenance. We discuss each in turn.

### 2.1 Semantic Inaccessibility

The complexity of systems in large, distributed organizations requires the decomposition of work into simpler tasks managed by smaller, more or less autonomous, teams within the larger organization. This results in the formation of localized task centric ontologies. This partitioning of the teams and tasks and its relation to the whole may be used to structure communication, information repositories, and lessons learned. Introducing new members to these teams require that the members learn the local ontology. In today's information-centered world, this manner of work distribution requires the *sharing* of information among the various contexts established by the different project teams and among the different tools supporting the activities performed by those teams. Hence, the usefulness of automated tools in this distributed environment is a function of the degree to which those tools can facilitate the sharing of information across different contexts for the agents that use them.

Early generation information systems adapted physical product information support concepts to these information product enterprises. However, the data produced by software of this kind is typically maintained in structured data sets and is ultimately grounded in a physical artifact concept. In many cases, each software tool has its own private data repository. The central problem with this is not necessarily that the *data*—i.e., the various forms of representation used in some aspect of an enterprise to carry information—generated by these tools is inaccessible. This problem is being addressed with varying degrees of success by technologies such as data warehousing and file exchange protocols. A more challenging problem is the fact that the semantic *content* of this data (i.e., the information itself) is not typically accessible to other (human or computer) agents in the organization. This latter problem is accentuated in information products organizations. That is, when the ultimate product of the organization is information (designs, requirements, plans, assessments, etc.), the links between the data in the enterprise system and the product can easily break down or, at best, become complex.

A complex representation (e.g., a data model or a business process model) carries the information in virtue of some established, systematic connection between the components of the representation and the world. It is this connection that determines the semantic content of the data being represented. Typically, however, the semantic rules of a representation system for a given application and the semantic intentions of the application designers are not advertised, or in any way accessible to, other agents in the organization. This makes it difficult, even impossible, for such agents to determine the semantic content of a database. We refer to this as the *problem of semantic inaccessibility*. This problem manifests itself superficially in the forms of unresolved ambiguity (as when the same term is used in different contexts with different meanings) and unidentified redundancy (as when different terms are used in different contexts with the same meanings).

## 2.2 Logical Disconnectedness

Even given a solution to the problem of semantic interpretability, however, an additional problem impedes full cooperation among disparate teams. Suppose, for instance, we have determined that a certain representation  $R1$  in a design model  $M1$  is semantically equivalent to a representation  $R2$  in a given analysis model  $M2$ , and that both  $R1$  and  $R2$  stand for the same entity. Thus, the models  $M1$  and  $M2$  both carry information about  $P$ . Suppose now that the information about  $P$  in  $M2$  is updated. This

requires a change in the information carried about  $P$  in  $M1$ . The fact that it is known that  $R1$  and  $R2$  are semantically equivalent in and of itself has no bearing whatever on whether the implications of the change in  $M2$  will be propagated to  $M1$ . More generally, then, a key problem is that the constraints between the particular pieces of information generated by various tools, both within and across enterprise contexts, are rarely maintained. We refer to this as the *problem of logical disconnectedness*.

## 2.3 Consistency Maintenance

Full integration, however, requires more than just the maintenance of constraints across enterprise domains. There is no guarantee that, for example, an application  $A1$  will remain internally consistent when it is updated in light of a change in another  $A2$ . More generally, a dynamic body of enterprise information is always subject to inconsistency. The simplest way in which this can occur is for a database to be updated with information that conflicts with information already in the database. The problem is exacerbated in a distributed environment, since inconsistency may arise not simply within a single database, but across two or more distinct databases: two databases  $B1$  and  $B2$  may both be internally consistent yet inconsistent with one another. This fact raises the question of how the source of inconsistency is to be detected and, once detected, managed within an integrated environment. The problem of inconsistency management is particularly important because inconsistency pollutes all information in a database and nothing that follows from such a database is reliable. Hence, mechanisms must be in place for isolating the sources of inconsistency in a set of related databases and returning the set to a stable, consistent state. The detection and management of inconsistency is referred to as the *problem of consistency maintenance*.

In summary, providing powerful tools for task-situated agents in information production require overcoming three major obstacles: (i) semantic inaccessibility, (ii) logical disconnectedness, and (iii) consistency maintenance. The approach described in this paper is designed to partially address these barriers.

## 3.0 Solution Concept Overview

The principal hypothesis of our approach is as follows: *semantic information sharing* and *information integration* for complex, dynamic, multi-organizational enterprises will be effectively enabled using an ontology-driven approach that automates the extraction and mediation of semantic information from unstructured text sources. Our approach uses a combination of knowledge discovery techniques with ontology matching and harmonization methods. Two

key capabilities are central to the success of our approach:

1. ontology extraction / revision from text data sources, and
2. knowledge integration to facilitate semantic information sharing and data mediation.

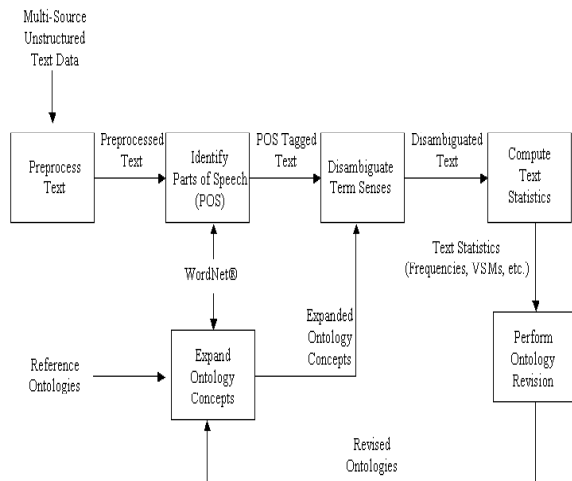
The intended initial end users of the tools based on this approach are systems analysts, knowledge engineers and, eventually, application subject matter experts.

### 3.1 Ontology Extraction / Revision From Text Data Sources

Our approach assumes the requirement to continuously generate/refine both local (task-centered) and context ontology concepts. It is assumed that no baseline (domain-specific) ontologies are available and that the subtle changes of even task specific ontology concepts, from task instance to instance, are relevant. In such a situation, the requirement exists to automatically extract/refine ontologies as a part of the process of analyzing the statistics and semantic information contained in the source information. It also requires providing a user-immersive means of presenting the findings and incorporating the user directives in the down stream processing.

#### 3.1.1 Ontology-Assisted Knowledge Extraction Approach

Our approach for performing such ontology-assisted knowledge extraction from data sources largely composed of unstructured text is summarized in Figure 1.



**Figure 1: Ontology-Assisted Semantic Information Extraction Method.**

### Processing Text

The raw text data from multiple sources (emails, reports, briefings, etc.) are first preprocessed. This activity includes parsing, format conversion (to the internal text representation format of the knowledge extraction tools), and filtering of non-value adding symbols and words.

### Identify Parts Of Speech (POS)

This activity identifies the Parts Of Speech (POS) for the terms in the document. The WordNet® lexical database is used to assist with this step. Note that a given word may have (i) multiple POS uses (e.g., the word ‘launch’ may be used as a noun or a verb); and (ii) multiple meanings for each POS use [e.g., the noun use of the word ‘launch’ may be a take-off (as for space vehicles at Kennedy Space Center) or a presentation (such as a sales briefing)]. POS tagging helps with the initial disambiguation relative to multiple meanings of a term, and interpreting the roles of the terms in relationships (e.g., noun – noun relations vs. noun –verb relations).

### Disambiguate Term Senses

This activity processes the pre-processed text along with the expanded ontology concept set (Figure 1). Term sense disambiguation narrows the scope of the possible meanings of a term and helps in extracting desired knowledge from text sources [1]. Techniques such as Hidden Markov Models (HMM) are used to facilitate term sense disambiguation.

### Compute Text Statistics

This activity extracts semantic information in text corpora (from multiple sources) using statistical methods. A key activity is to transform the term and concept information contained within the expanded ontology and multiple text data sets to a form that would facilitate the application of statistical analysis and machine learning techniques. Multiple statistical representations have been investigated, including (i) Term and Concept Frequency Counts, (ii) Term Vector Space Models (T-VSMs), and (iii) Concept Vector Space Models. We will explain the notion of a Vector Space Model (VSM) and then outline T-VSMs and C-VSMs. VSMs are widely used to determine the relative importance of words in document collections and to assist in the analysis of the semantic context in large text collections. The VSM creates a space in which both documents and terms are represented by vectors. For a fixed collection of documents, an m-dimensional vector of associated weights is generated, where m is the number of unique terms in the document collection. A *Term VSM* (T-VSM) is a model using the terms and concepts in a document

without reference to the concepts and terms of the expanded ontology. The *Concept VSM* (C-VSM) conditions the weights of the T-VSM by measuring the degree of match between the terms and concepts in the expanded ontology. Intuitively, the higher the number of common words between the ontology and a document, the higher the value of the C-VSM vector.

The text statistics produced by the above methods are used in the ontology comparison analysis used for knowledge mediation (Section 3.2).

### Perform Concept Discovery

The main result of the concept discovery process is a set of candidate “enhancements” to the reference domain ontologies. The ontology enhancements will include candidate new classes (types), characteristics (properties), relations (associations), and axioms (“sanctioned inferences”).

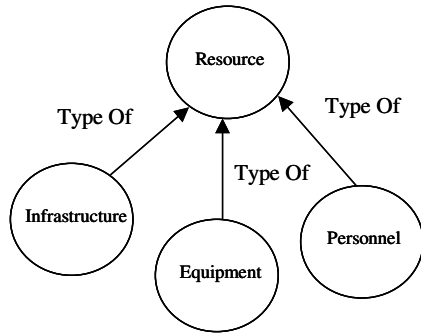


Figure 2: Example Resource Ontology Fragment.

Table 1: Expanded ontology concepts.

Parent Term	Subordinate Term	Weight
Personnel	Human Resource	90
Personnel	People	85
Personnel	Staff	80
Equipment	Machine	95
Equipment	Tool	90
Equipment	Instrument	80
Equipment	Apparatus	50
Infrastructure	Communication	80
Infrastructure	Building	80
Infrastructure	Road	70
Infrastructure	Transport	65

### Expand Ontology Concepts

The reference ontologies are represented in the standard Web Ontology Language (OWL) format. The ontology expansion activity generates all synonyms of the ontology terms (types and relations; will include multi-word terms such as “Part Of,” “Composed Of,” “Greater Than,” etc.). The intent of this activity is to produce a collection of words that (approximately) represents the semantics entailed by the (abstract) information encodings that are contained

in a structured ontology. The expanded ontology concepts are used to assist in the interpretation and analysis of the text contained in the unstructured text data sources. An example of ontology expansion is illustrated in Figure 2 and Table 1. The term “Weight” signifies, in a relative sense, the strength of the association between the parent term and the subordinate term.

### 3.1.2 Automatic Ontology Extraction from Text Corpora

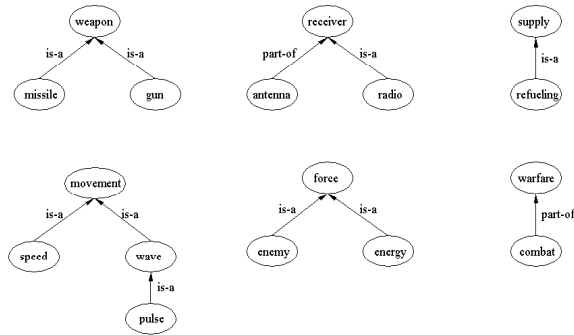
As with many other machine learning and natural language processing (NLP) applications, the biggest challenges of ontology extraction are speed and accuracy. Automatic ontology extraction has three core tasks: concept extraction, taxonomy extraction (i.e., IS-A), and non-taxonomic relation extraction (i.e., PART-OF). The core tasks might require other subtasks including text parsing, POS tagging, ontology mapping, pruning, and merging. The components performing these tasks govern the overall speed of the ontology extraction process. Among these tasks, non-taxonomic relation extraction is the most difficult because (i) the same relation may appear in different forms in free text, and (ii) the same term may have different meanings in different contexts. Researchers have tried to use WordNet® to determine the semantic sense of terms; however, the overhead costs attributed to calling WordNet® and a POS tagger can be cost prohibitive.

The issue of accuracy is even more challenging than in its machine learning and NLP counterparts because (i) no standard evaluation measures, like precision and recall, are available to estimate the quality of an ontology extraction technique, and (ii) the reference ontologies, which are the “ground truth” for evaluating the extracted ontologies, are subjective with regard to the human experts who prepare them. The performance of POS taggers also varies and can be error prone [2].

We have developed two approaches for automatic ontology extraction. The first approach uses WordNet® for tagging POS and a Hidden Markov Model (HMM)-based approach for POS disambiguation. It is capable of extracting two-place collocations such as *adjective-noun*, *noun-noun*, *noun-preposition-noun* (proper nouns are treated as nouns). Examples of some of the collocations extracted are *weapons of mass destruction* (*noun-preposition-noun-noun*), *cease fire agreement* (*noun-noun-noun*), *Islamic brotherhood movement* (*adjective-noun-noun*).

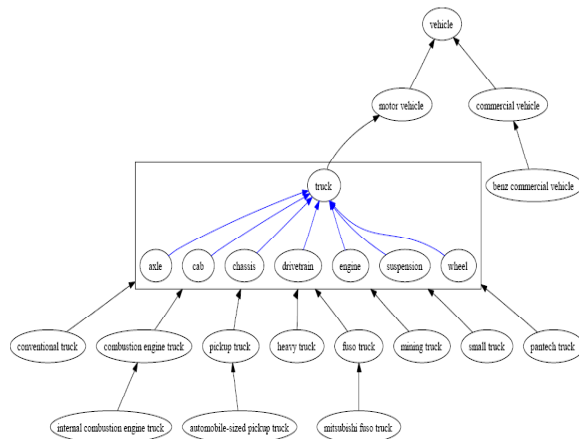
Since the number of candidate concepts can be very high, concepts occurring less than the specified threshold frequency are filtered out. Relationships

between concepts and attributes are discovered using WordNet®. Figure 3 shows an example of ontology extracted from an air-to-air combat related document.



**Figure 3: Example of an Extracted Ontology Extracted Using WordNet®.**

Relying on WordNet® to extract domain-specific knowledge is sometimes difficult since WordNet® includes general terms (more than 120,000 words) but very few domain-specific terms. To compound the matter, a term in a given domain may have a totally different meaning from one given by WordNet®. For example, OWL stands for Web Ontology Language in a semantic web context, but its only meaning in WordNet® is a kind of bird.



**Figure 4: Example of an Extracted Ontology Without Using WordNet®.**

To overcome the shortcomings of WordNet®, we have developed another approach for extracting ontologies without using WordNet®. It has a built-in POS tagger that can process tens of thousands of words in one second. This approach is capable of extracting concepts and identifying taxonomy (IS-A) and non-taxonomic (PART-OF) relations. We have demonstrated this approach using a collection of documents from Wikipedia®. It creates ontologies from the Web pages instantaneously, and has the potential to search linked pages to refine the extracted

ontology. Figure 4 shows an extracted ontology from a Web page describing trucks. The edges outside the box represent IS-A relations and other edges are for PART-OF relations. This approach has successfully identified different categories of vehicles, such as a truck, various types of trucks, and key components of a truck. Our implementation can present extracted ontologies in various formats by exporting them into an OWL-compatible XML format (as shown in Figure 3) or into DOT scripts [3] that can be easily plotted in various image formats (see Figure 4).

### 3.2 Knowledge Integration to Semantic Information Sharing and Data Mediation

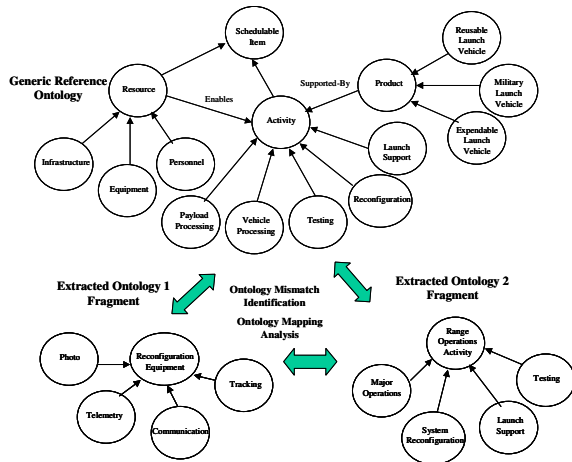
Knowledge mediation methods are used to (i) identify ontology mismatches (between the extracted ontology concepts and the reference ontologies) and (ii) generate ontology mapping advice (conflict resolution, ontology alignment and merging). Our ontology mismatch analysis approach is summarized in the following paragraphs.

The identification and analysis of ontology mismatches will provide key insights that facilitate information sharing, integration, and communication in distributed heterogeneous environments. The ontology mismatch analysis will assess many different types of mismatches that have been found to inhibit the integration / interoperability of heterogeneous systems.

Our ontology mismatch analysis approach identifies ontology similarities based on different dimensions or aspects of an ontology as outlined below (see [4] also).

1. Terminology: This analysis finds similarities between concepts based on the words used in naming the concept.
2. Feature Based: This analysis finds similarities between concepts based on the attributes and attribute values.
3. Topology: This analysis finds similarities between concepts based on the structure (i.e., based on relationships the concept has with other concepts).
4. Semantics: This analysis finds similarities between concepts based on the semantic analysis of “domain discourses or descriptions” (i.e., by semantic analysis of statements made by domain experts about the focus concepts. The corpora of relevant domain documents, and even textual descriptions by experts, are analyzed in this method to determine the true meaning of the

concepts and the manner in which domain experts are using them in practice.



**Figure 5: Ontology Mismatch and Mapping Operations Scheduling Application Example.**

Our research experience indicates that the similarity analyses based on some of the aspects are not consistent with others. For example, mappings based on terminology might not agree with topology. This is especially true between terminology and other analyses methods, because different domains are very likely to use different words to refer to the same concepts. Perhaps, more problematically, different domains are likely to use the same words with different semantics. Another aspect of our approach is to analyze the nature of similarity mappings, so as to determine the inconsistency between the various techniques. The inconsistency analyses give valuable insights into the manner in which different domains use the terminology, features, relations, or concept meanings differently (Figure 5).

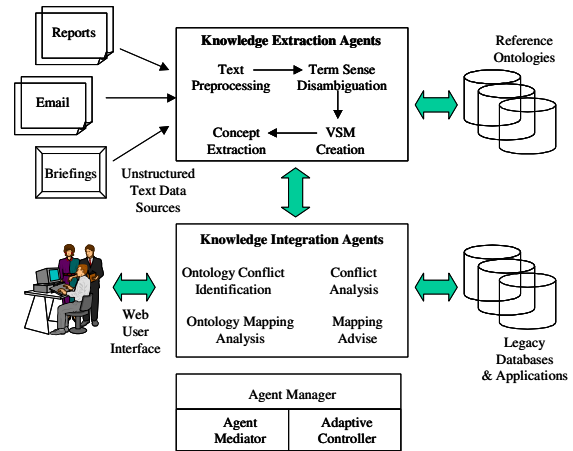
Finally, the similarity values obtained by various algorithms are fused using data and information fusion approaches so as to generate comprehensive, aggregate similarity mappings between concepts belonging to different ontologies.

The key steps in our mismatch analysis approach are outlined below:

1. Determine the mappings between concepts using different (orthogonal) metrics like nomenclature, attributes, topology, or structure and the manner in which concepts are used in the domain documents and disclosures.
2. Analyze the inconsistencies of mappings based on different metrics. For example, concepts that have similar attributes can be named differently.
3. Fuse the results to obtain conflict resolved mappings (see [5] also).

## 4.0 Architecture

A preliminary architecture to provide automated support for our approach is shown in Figure 6.



**Figure 6: Conceptual Architecture - Ontology Driven Integration.**

Two key subsystems of this architecture assume an *agent-based approach* in their design: (i) the Knowledge Extraction Agents (KEA) and (ii) Knowledge Integration Agents (KIA). The *Agent Manager (AM)* mediates, coordinates, and controls the tasks performed by the KEA and KIA agents. The main components of the AM are the Agent Mediator and the Adaptive Controller. The Agent Mediator is a registry of service seeking agents and service providing agents. All agents will be aware of the location of the Agent Mediator and can communicate with the Agent Mediator. All service providers, upon their activation, will register their services with the mediator. The Adaptive Controller (AC) provides innovative mechanisms for the agents to monitor and improve their own performance over extended periods of time. The purpose of the AC is to both detect problems early and correct them, and to induce knowledge extraction and integration process improvement over time. The functionality of the other subsystems and components are now outlined.

The Web user interface is designed to allow end users who are distributed in time and space to access and collaborate in knowledge sharing and integration activities. The user interface allows for customization to the unique needs of multiple end user groups.

The Knowledge Extraction Agents (KEA) provide automated support for the knowledge extraction process described in the previous section. The functions provided by KEA include (i) Text Pre-Processing, (ii) Term Sense Disambiguation, (iii) Text Statistics Generation, (iv) Concept Discovery, (v) Ontology Mismatch Analysis (Conflict Identification

and Conflict Analysis between the extracted ontology and the reference ontology), and (vi) Ontology Mapping Assistance (Ontology Alignment and Merging).

The ontologies generated in support of specific task instances are cataloged as reference ontologies to be used as the basis for interpreting and mediating the information extracted from multiple text-based data sources in future tasks.

## 5.0 Summary

This paper described the motivations, approach, and architecture to use ontologies for knowledge extraction and for applications that assist situated agents in complex information integration tasks. We described ontology-driven information extraction and knowledge mediation methods to support disciplined information integration. Finally, we described an architecture that provides automated support for ontology driven integration. We are currently designing and building multiple applications that validate the practical benefits of this research.

## 6.0 References

- [1] E. Agirre and D. Martinez, “Knowledge sources for word sense disambiguation,” *Lecture Notes in Computer Science*, vol. 2166, 2001.
- [2] M. KAVALEC and V. SVÁTEK, “A study on automated relation labeling in ontology learning,” in *Ontology Learning From Text: Methods, Evaluation and Applications*, P. Buitelaar, P. Cimiano, and B. Magnini, eds., Washington, DC: IOS Press, 2005.
- [3] E. R. Gansner and S. C. North, “An open graph visualization system and its applications to software engineering,” *Software Practice and Experience*, vol. 30, no. 11, pp. 1203–1233, 2000.
- [4] M. Klein, “Combining and relating ontologies: an analysis of problems and solutions,” in *Workshop on Ontologies and Information Sharing, IJCAI’01* pp. 53-62, 2001.
- [5] D. McGuinness, R. Fikes, J. Rice, and S. Wilder, “An environment for merging and testing large ontologies,” in *Proc. of the 7th International Conference on Principles of Knowledge Representation and Reasoning*, pp. 483–493, 2003.