

Ontology-based Semantic Classification of Unstructured Documents

Ching Kang Cheng¹, Xiaoshan Pan² Franz Kurfess¹

¹Department of Computer Science California Polytechnic State University
San Luis Obispo, California 93407, USA
{fkurfess, ckcheng}@calpoly.edu

²Department of Civil and Environmental Engineering Stanford University
Stanford, California 94305, USA
xpan@stanford.edu

Abstract. As more and more knowledge and information becomes available through computers, a critical capability of systems supporting knowledge management is the classification of documents into categories that are meaningful to the user. In a step beyond the use of keywords, we developed a system that analyzes the sentences contained in unstructured or semi-structured documents, and utilizes an ontology reflecting the domain knowledge for a semantic classification of the documents. An experimental system has been implemented for the analysis of small documents in combination with a limited ontology; an extension to larger sets of documents and extended ontologies, together with an application to practical tasks, is the focus of ongoing work.

1.0 Introduction

With the volume of knowledge and information available to computer user increasing at an ever-accelerating rate, the need for an effective mechanism to organize not only information, but also knowledge becomes critically important. We distinguish knowledge from information and define knowledge as “fluid mix of framed experience, values, contextual information and expert insight that provide a framework for evaluation and incorporating new experiences and information” [9]. Information retrieval techniques such as document clustering techniques have been employed frequently to support the organization and retrieval of information [1]. Document clustering is essentially an unsupervised process where a large collection of text document is organized into groups of documents that are related, without depending on external knowledge [10]. A potential problem with the data-driven clustering algorithms is the inability to correctly identify cases when different words are used to describe the same concept. This is due to the similarity-based measure adopted in the algorithm. Furthermore, without including the user context, more often than not, information is organized according to the fixed

viewpoint of the conventional clustering methods, rather than reflecting the interests of the user [1]. This will ultimately discount the usefulness of the information. Typically, such retrieval techniques leave a significant portion of the utilization of knowledge contained in the retrieved documents to the user: these techniques are only used to calculate a ranking of the documents, attempting to identify the ones that are most relevant to the user. The core principle of our approach is based on our belief that information has to be organized in a manner that is intuitive and relevant to the user to be useful. A context model is the ontology of an application domain, which defines the meanings of vocabularies used within the ontology according to the user perspective. The context model attempts to emulate the mental processes of perception and categorization. We have developed the Ontology-based Semantic Classification (OSC) framework, leveraging on natural language processing techniques and ontologies to incorporate the user current context into the categorization of information. Figure 1.0 illustrates the overall process where unstructured documents are categorized according to the user perspective. In Section 2, we discuss the usage of ontology in the OSC framework. Section 3 presents the various components employed. In Section 4, we show the implementation as well as the results of the experiments performed. In Section 5, we summarize our findings and future endeavor.

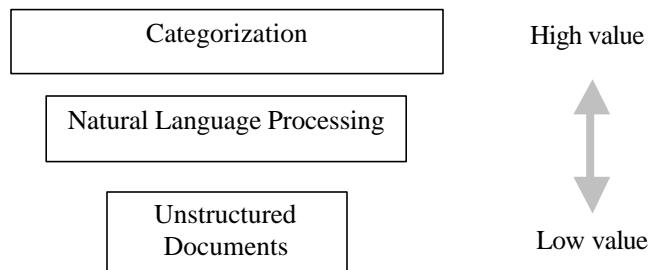


Fig. 1. Overall classification process

2.0 Ontology

An ontology can be defined as specification of a representational vocabulary for a shared domain of discourse which may include definitions of classes, relations, functions and other objects [2]. An ontology includes a selection of specific sets of vocabulary for domain knowledge model construction, and the context of each vocabulary is represented and constrained by the ontology. Therefore, an

ontological model can effectively disambiguate meanings of words from free text sentences, overcoming the problem faced in natural language where a word may have multiple meanings depending on the applicable context [3]. Vocabularies used in an ontology are two kinds: 1) a direct subset of a natural language (e.g., "entity", "tree", and "basketball"), and 2) user-created 'words' that does not exist in an natural language (e.g., "ALLFRD"). Depending on the construction of the ontology, the meaning of those words in the ontology could remain the same as in natural language, or vary completely. The meaning of ontological terms that are not derived directly from a natural language can still be captured by a natural language. For example, the word "COM" used in a specific ontology means "Common Object Model" in English. From an engineering perspective, ontologies can be very helpful with the reuse of domain knowledge, and for the separation of domain knowledge and software code that performs operations on that knowledge. We have adopted ontologies as the link to incorporate user-specific context into the categorization process within the framework. Essentially, within the Ontology-based Semantic Classification (OSC) framework, the context model is represented by the *signature* and *category* ontology. Signature ontology is a logical grouping of keywords having the same meaning. For example, the signature SEARCH contains the keywords "A*", "Depth First" and "Breadth First". Signature ontology is used in Context-based Free Text Interpreter (CFTI) to extract signatures from unstructured documents. This is the first stage of the categorization process. The category ontology is a higher level of logical grouping used in the next stage. A category contains signatures with the same context. For example, the category AI is a logical grouping of the signatures SEARCH and AGENT. Context-based Categorization Agent (CCA) employs the category ontology to categorize the signature instances extracted from the unstructured documents.

3.0 Semantic Classification

Linguistically, humans combine understanding of relatively small textual units in order to understand larger textual units, guided by syntactic and semantic rules [11][12]. Syntax relates to arrangement, and semantic to the meaning of words. Similarly, it is necessary for a natural language processing system to be able to address syntactic and semantic aspects of natural language [3]. Subsequently, to perform useful classification, the categorization must be based on the actual information content or explicit representation of the information content of the source documents. In this section, we introduce two existing language tools (i.e., Link Grammar Parser and WordNet), and the design of Context-based Free Text Interpreter (CFTI) and Context-based Categorization Agent (CCA).

3.1 Syntactic Analysis

Natural language syntax affects the meaning of words and sentences. The very same words can have different meanings when arranged differently. For example: “a woman, without her man, is nothing” and “a woman: without her, man is nothing” (<http://www.p6c.com/joke of the week.html>). The Link Grammar Parser, developed at Carnegie Mellon University [4], assigns to a given sentence a valid syntactic structure, which consists of a set of labeled links connecting pairs of words. It utilizes a dictionary of approximately 60,000 word forms, which comprises a significant variety of syntactic constructions, including many considered rare or idiomatic. The parser is robust; it can disregard unrecognizable portions of sentences, and assign structures to recognized portions. It is able to intelligently guess, from context and spelling, probable syntactic categories of unknown words. It has knowledge of capitalization, numeric expressions, and a variety of punctuation symbols. The basis of the theory of Link Grammar is planarity, described by [5], as a phenomenon evident in most sentences of most natural languages. To represent a sentence, arcs are drawn connecting words with specified relationships within sentences. These arcs do not cross for syntactically correct sentences. Planarity is defined in Link Grammar as “the links are drawn above the sentence and do not cross” [4]. To visualize link grammars, think of words as blocks with connectors coming out. There are different types of connectors; connectors may also point to the right or to the left. A sentence is valid if all the words present are used according to their rules, and certain global rules are satisfied [6]. Each word is a block with connectors (see Figure 2).

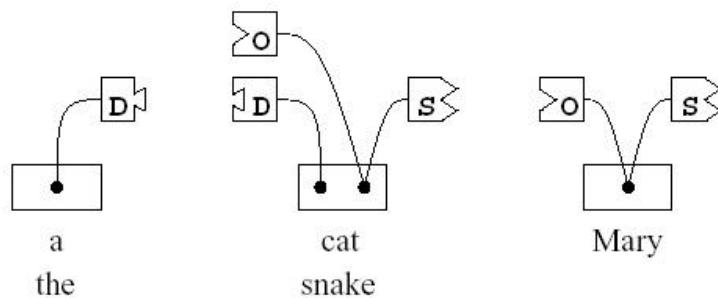


Fig. 2. Each word is a block with connectors [6]

Each intricately shaped, labeled box is a connector. A connector is ‘satisfied’ when ‘plugged into’ a compatible connector (as indicated by shape). A valid sentence is one in which all blocks are connected without a crossing. An example of a valid

sentence is “the cat chased a snake” (Figure 3). An example of an invalid sentence is “the Mary chased cat”, which contains a cross (Figure 4).

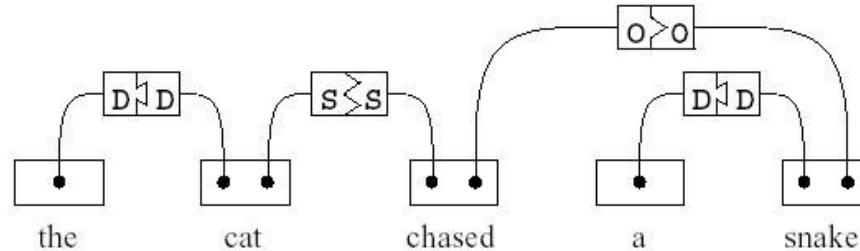


Fig. 3. A valid sentence contains blocks connected without a cross [6]

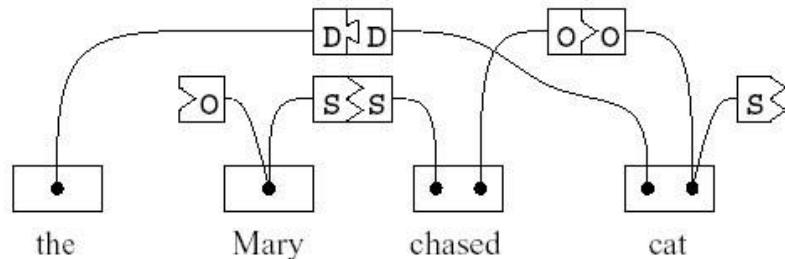


Fig. 4. An invalid sentence contains blocks connected with crosses [6]

The Link Grammar Parser identifies all valid linkages within a free text input, and outputs them as grammatical tree. For example, an input such as “The brown fox jumped over that lazy dog” would result in the output shown in Figure 5:

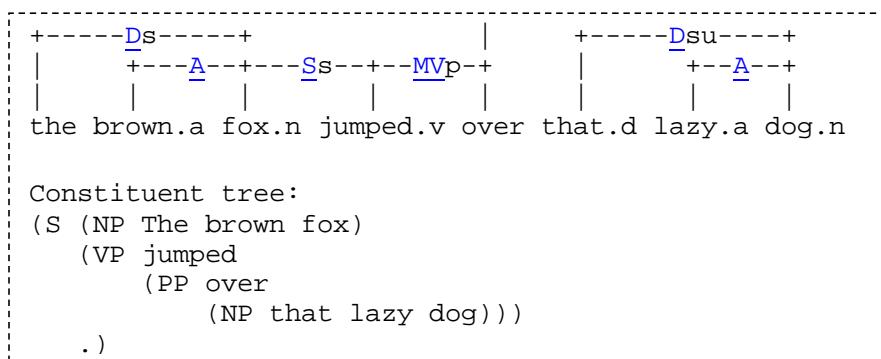


Fig. 5. An output produced by the Link Grammar Parser

3.2 Semantic Knowledge

Two types of semantic knowledge are essential in a natural language processing system: Lexical knowledge among words independent of context (e.g., “children” as the plural form of “child”, and the synonym relationship between “helicopter” and “whirlybird”) and contextual knowledge (i.e., how meanings are refined when used in a specified context). In Context-based Free Text Interpreter (CFTI), lexical knowledge is acquired through integration of the system with the WordNet database, and contextual knowledge is acquired by tracking contextual meanings of words and phrases during and after development of an ontology (i.e., context model). *WordNet*, an electronic lexical database, is considered to be the most important resource available to researchers in computational linguistics, text analysis, and many related areas [7]. WordNet has been under development since 1985 by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller. Its design is “...inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, and adjectives are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets.” [8]

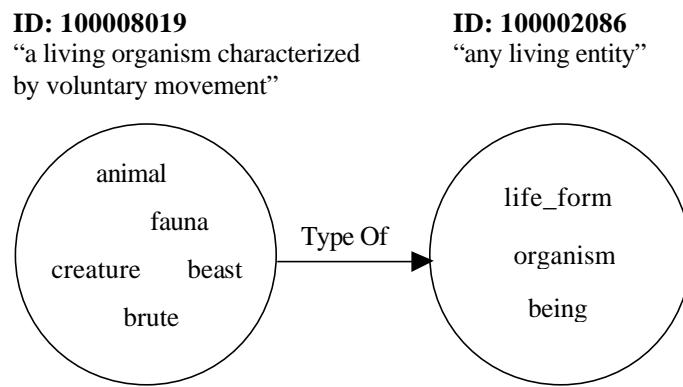


Fig. 6. Two synsets with a ‘type-of’ relationship

The most basic semantic relationship in WordNet is synonymy. Sets of synonyms, referred to as synsets, form the basic building blocks. Each synset has a unique identifier (ID), a specific definition, and relationships (e.g., inheritance, composition, entailment, etc.) with other synsets. Two synsets with a “type-of” relationship are shown in Figure 6. The first synset has an ID “100008019”, a definition of “a living organism characterized by voluntary movement”, and

contains six individual words (e.g., "animal", "animate being", etc.). The second synset has an ID "100002086", a definition of "any living entity", and it contains three words (e.g., "life form", "organism", and "being"). The first synset is a "type-of" the second synset. WordNet contains a significant amount of information about the English language. It provides meanings of individual words (as does a traditional dictionary), and also provides relationships among words. The latter is particularly useful in linguistic computing. While WordNet links words and concepts through a variety of semantic relationships based on similarity and contrast, it "does not give any information about the context in which the word forms and senses occur" [7]. In Context-based Free Text Interpreter (CFTI), refinement of word meanings in specific contexts (i.e., contextual knowledge) is accomplished by mapping relationships between natural language and a context model represented by the signature ontology. In practice, the tracking of mapped relationships between a natural language sentence and a context model is a process of interpretation of the model (i.e., what a model really means) through the use of a natural language. From the perspective of a natural language processing system, which employs appropriate lexical and contextual knowledge, the interpretation of a free text sentence is a process of mapping the sentence from natural language through a context model (Figure 7).

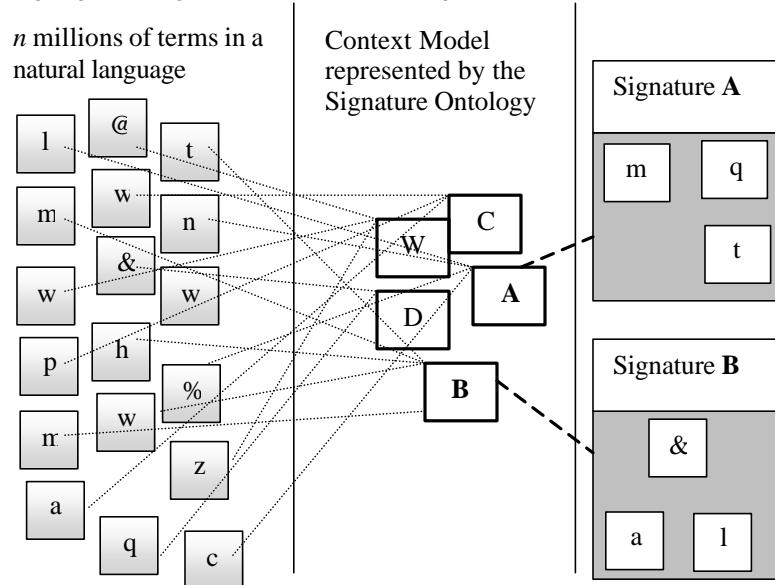


Fig. 7. Mapping from natural language to signature instances through signature ontology (context model)

Different context models may produce different results simply because words can have different meanings in different contexts. In Context-based Free Text Interpreter (CFTI), the representation of meaning is accomplished by manipulations of a context model (i.e., creation, modification, and deletion of objects and relationships in the signature ontology). For example, a hazard detection system receives a free text sentence "House 303 is on fire!". If the system is able to model this information correctly (i.e., locate the instance of the house in the model and set its attribute to "on fire"), then it is assumed that the system understands the meaning of the sentence [3].

3.3 Context-based Free Text Interpreter (CFTI) Design

CFTI leverages on the Link Grammar capability for syntactical analysis of a sentence. At the same time, the lexical meaning analysis of a sentence is supported through the integration with the WordNet database [3]. The tasks performed by CFTI are summarized as follows: 1). Analyze the syntactic structure of the sentence. 2). Analyze the lexical meaning of the words in the sentence. 3). Refine the meanings of the words through the application of a signature ontology (context model). 4). Represent the meaning of the sentence in signature instances. Figure 8 illustrates the processing of a free text message by the CFTI system and the subsequent representation in the signature instances.

Even though the CFTI requires an ontological model for the acquisition of contextual knowledge and the representation of meanings, the system is not constrained by any particular knowledge domain. A system change from one ontological model to another does not require significant system reconfigurations.

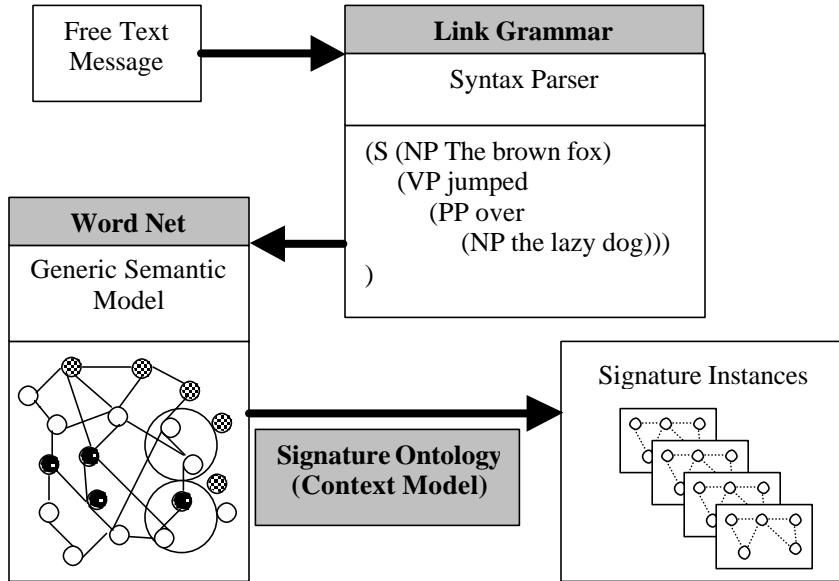


Fig. 8. From free text messages to signature instances

3.4 Context-based Categorization Agent (CCA) Design

The signature instances produced from Context-based Free Text Interpreter (CFTI) correlate the content of the unstructured documents with the context of the user. A signature is defined as a logical grouping of keywords having the same context in the user perspective. For example, in the user perspective, keywords such as "A*", "Depth First" and "Breadth First" sharing the same context can be grouped under the signature SEARCH. CCA offers flexibility and scalability by providing a higher level of grouping: signatures with the same context are grouped in the same category. CCA relies on the *category ontology* to incorporate the user perspective. The category ontology specifies how the signatures are grouped with reference to the user context. For example, if the signatures SEARCH and GAME share the same context, they can be grouped in the same category. In addition, the category ontology supports class hierarchy similar to the object-oriented paradigm, where parent child relationship exists. A key feature of such an approach is the capability to adapt to changes dynamically, without recompilation of the CCA. This is especially important since changes had to be made to the category ontology frequently, to reflect the changes in the user perspective, which evolves as the volume of information increases. For example, modification is made to the

category ontology as new categories are created or new signatures are added to existing categories. CCA provides the flexibility and scalability to adapt to these changes without any recompilation. The tasks performed by CAA are: 1). Interface with the signature instances. 2). Interface with the category ontology. 3). Classify the signature instances through the application of category ontology. 4). Represent the classification of the documents as category instances. Figure 9 illustrates the classification of the signature instances by the CAA and the subsequent representation.

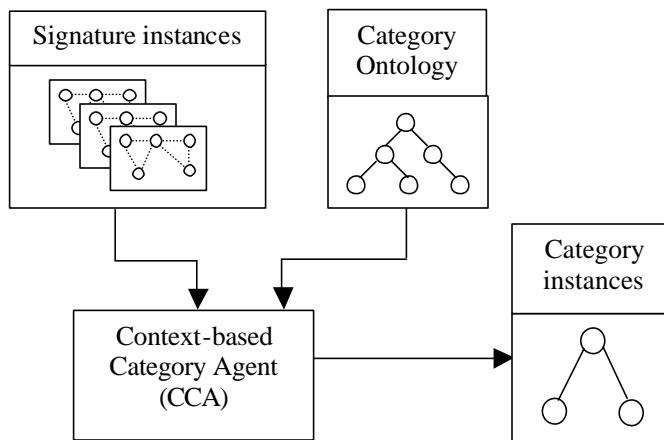


Fig. 9. From signature instances to category instances through category ontology

4.0 Implementation

This section explains a prototypical implementation of an ontology-based system for the semantic classification of unstructured documents. We demonstrate the feasibility of incorporating user context for the task of classifying unstructured documents.

4.1 Ontology-based Semantic Classification (OSC) Framework

The core design principle of the OSC framework is to provide loosely coupled yet seamlessly integrated components. To achieve this, the OSC framework architecture is decomposed into three distinct layers and the interfaces between the components are specified in a language neutral format (e.g. via XML), as shown in Figure 10.

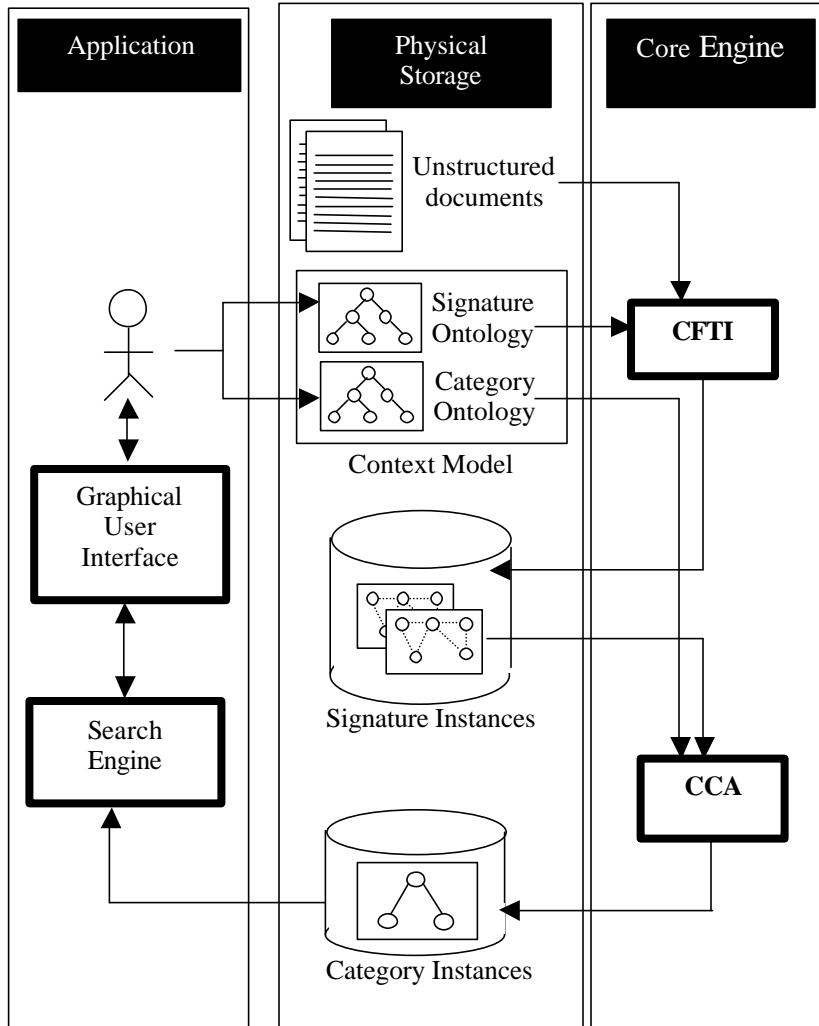


Fig. 10. System Architecture of the OSC Framework

The *core engine layer* encompasses components that contribute to the core functionality of the framework. It includes the Context-based Free Text Interpreter (CFTI) and Context-based Categorization Agent (CCA). CFTI is implemented through the use of CLIPS 6.20. CFTI contains five components: Link Grammar, Lisp Simulator, WordNet, a mapping engine, and a signature ontology (context model). The Link Grammar and Lisp Simulator process syntactic knowledge;

WordNet provides lexical knowledge about words; the mapping engine is composed of CLIPS rules for meaning extraction from free text sentences; and the context model provides contextual knowledge about words and representation of meanings of free text sentences [3]. While a context model is required by the system, a change from one context model to another does not require significant system reconfiguration. CCA was developed in CLIPS 6.20. It includes two components: a classification engine and a category ontology. The classification engine is powered by a network of rules that categorizes the signature instances with respect to the interest of the user as specified in the category ontology. The category ontology can be extended dynamically to allow changes without recompiling the system. The *physical storage layer* handles the storing of the context model represented by the signature and category ontologies, signature and category instances and the unstructured documents. The interface between the physical layer and the rest of the components is confined to a language neutral format such as XML, ensuring the loose coupling between the different layers. Applications that have to interact with the physical layer can be written in any programming language as long as that language supports XML. On the other hand, the signature and category ontologies (context model), signature and category instances and the unstructured documents can be stored in text file format, binary file format, relational database and object-oriented database. The *application layer* is a logical grouping of components that capitalize on the category instances. By design of the OSC framework, application components can be plugged into the framework as and when they are ready. A possible application component is the search engine. The search engine allows the user to retrieve relevant information from the unstructured documents using the category ontology as search criteria.

4.2 Experiment Results

The main purpose of the experiment is to verify that the Ontology-based Semantic Classification (OSC) framework functions as a whole. This includes accessing how well the overall system performs when the individual components are integrated together through the common interfaces. A secondary objective of the experiment is to validate the usefulness of the purposed context model in emulating the categorization process of the human being. For this experiment, we chose 33 unstructured documents from the American Association for Artificial Intelligence web site (www.aaai.org) in various categories. Each document was subsequently converted to the text format. The experiment was performed in two stages. In the first stage, a human operator (a domain expert) was asked to manually categorize the documents into 5 distinct categories, Agent, Games, Data Mining, Natural Language Processing and Search. There was no restriction to the number of category that a document could be categorized. In the second stage, the OSC framework categorized the same collection of documents. This involved a

knowledge engineer interviewing the domain expert to capture and represent his context into the context model used in the OSC framework. The result of the experiment is tabulated in Table 1.

DOCUMENTS			HUMAN					OSC Framework				
	Title	Pages	A	G	D	N	S	A	G	D	N	S
1	HeuristicSearch	7				x						x
2	ProblemSolvingSearch	6				x		x				x
3	PerformanceEvaluation	27			x						x	
4	EmpiricalMethods	25			x				x	x		
5	NLUnderstanding	11			x						x	
6	PresidentialAddress	8			x						x	
7	LipsToPrinter	5			x						x	
8	ComputationalLinguistics	2			x						x	
9	PlayingWithAI	37	x			x	x	x	x	x	x	x
10	GamutOfGames	23	x			x		x	x	x		x
11	TheNextMoves	19	x	x		x	x	x		x	x	x
12	ChampionsOfGame	9	x					x	x			x
13	PlayingYourCardsRight	6	x						x			
14	DoNotPassGo	4	x						x	x		
15	NeuralNetBackgammon	3	x						x			
16	ManVsMachinePlaysOut	2	x						x			
17	AncientGameGetsNewLife	1	x						x			
18	MatchingChessWits	1	x									
19	KnowledgeDiscoveryInDB	6		x			x		x			
20	ExploitingHiddenTrends	5		x						x		
21	MiningForTrendsAtHelpDesk	1		x						x		
22	ArchitectureForAgent	15	x					x	x		x	
23	AgentsAsEmbodiedAI	16	x					x			x	
24	ReduceWorkInfoOverload	15	x					x			x	
25	MultiAgentFrameworks	12	x					x			x	
26	AgentOrProgram	11	x					x	x	x	x	
27	AgentsCreativity	9	x					x				
28	AgentInFuture	7	x					x		x	x	
29	AgentsOfCooperation	7	x					x				
30	StagingAComeback	6	x					x				
31	ManyFacesOfAgents	8	x					x				
32	EaseBurdens	4	x					x				
33	IDA	2	x					x				

Table. 1. Experiment Result

The table shows that the system correctly identified most categories for the document, but in many cases, it selected multiple categories. In the terminology of information retrieval, recall was excellent, but precision was not so good. The key emphasis of this experiment, however, was not so much on the accuracy of the OSC framework, but rather to prove that the respective components can be integrated

and work seamlessly as intended. The OSC framework indeed performed as intended throughout the experiment, categorizing documents with respect to the context specified in the ontology. Further experiments with larger data sets and more finely tuned parameters are in process.

5.0 Conclusion

In this paper, we have shown how to utilize user context and preferences through ontology in order to classify unstructured documents into useful categories. We have demonstrated the use of a Context-based Free Text Interpreter (CFTI), which performs syntactical analysis and lexical semantic processing of sentences, to derive a description of the content of the unstructured document. Direct and indirect mapping relationships exist among vocabularies used by ontologies and vocabularies used by natural languages. The capture and utilization of these relationships is key to the development of natural language processing systems. The quality of classification of unstructured document is strongly dependent on the quality of context models and the accuracy of the interpretation of natural language. The Ontology-based Semantic Classification (OSC) framework has been tested with a relatively small context model. While an assumption that the system would perform similarly when tested with larger-sized models seems valid, conducting such tests is the focus of ongoing work, together with the use of the OSC in practical applications.

References

1. Kim, H.J., Lee S.G.: A semi-supervised document clustering technique for information organization. Proceedings of the ninth international conference on Information and knowledge management. McLean, Virginia (2000)
2. Gruber, T.: A translation approach to portable ontology specifications. Knowledge Acquisition, An International Journal of Knowledge Acquisition for Knowledge-Based Systems, 5(2). June (1993)
3. Pan, X.S.: A context-based free text interpreter. California Polytechnic State University San Luis Obispo Master's Thesis - Computer Science Department. Aug (2002)
4. Sleator, D., and Temperley, D.: Parsing English with a Link Grammar. Carnegie Mellon University Computer Science technical report CMU-CS-91-196. (1991)
5. Melcuk, I.: Dependency Syntax: Theory and Practice. New York: State University of New York Press. (1988)
6. Temperley, D., Sleator, D., and Lafferty, J.: An Introduction to the Link Grammar Parser. Technical report, Available: <http://www.link.cs.cmu.edu/link/> March 1999
7. Fellbaum, C.: WordNet: An Electronic Lexical Database. Cambridge: MIT Press. (1999)
8. Miller, G.: Wordnet: An Online Lexical Database. Int'l J. Lexicography, Vol. 3, No. 4, 1990, pp. 235-312. (1990)
9. Hahn, J., Subramani, M. R.: A framework of knowledge management systems: issues and challenges for theory and practice. Proceedings of the twenty first international conference on Information systems (December 2000)
10. Kim, H.J., Lee S.G.: A.I. and computational logic: An effective document clustering method using user-adaptable distance metrics. Proceeding of the 2002 ACM symposium on Applied computing (March 2002)
11. Minsky, M.: The society of Mind. New York: Simon and Schuster. (p.266). (1985)
12. Cole, R., Mariani, J., Uszkoreit, H., Varile, G., Zaenen, A., and Zampolli, A.: Survey of the State of the Art in Human Language Technology. Cambridge: Cambridge University Press. (p. 109) (1998).