

Natural language reasoning through model-based engineering

Gustavo Cabral and Tetsuo Tamai
{gustavo,tamai}@graco.c.u-tokyo.ac.jp

General System Studies
Graduate School of Arts and Sciences
The University of Tokyo - Japan

1 Introduction

The automatic processing of natural language written artifacts with the purpose of validation or interpretation (reasoning) about definitions and fact is a challenging goal. This task is divided here in several sub-tasks that are naturally accomplished by ourselves when reading and analysing texts containing information about any domain. Some of these tasks correspond to lexical, phrasal structure, grammatical relation analysis, models creation, validation and transformation, since this paper presents a model-based approach.

Belief Revision

In the belief revision or belief change areas, theoretical frameworks are defined to handle rational changes of belief, focusing on revisions that occur when one receives new information that is possibly inconsistent with the present state of belief. This type of knowledge construction is adopted here for the definition of a reasoning strategy that used model-based tools to implement any necessary task previously cited. In simple words, the state of a reasoner is defined by a set of logical predicates that contains precisely the meaning extracted from the processed sentences from a text, which is initially believed to be true. The incorporation of new information depends on the consistency of the existing knowledge base when updated with new beliefs.

The belief revision approach focus on maintaining a consistent knowledge base that is used for construction of logical conclusions based on facts. We must also accept all the logical consequences of the new belief. If the new information is inconsistent with the current state of belief, then some of the old beliefs must be retracted in order to preserve the consistency of the belief state. Therefore, the reasoner's belief state can be changed through three types of belief changing operations: Expansion (to add new beliefs without retraction of any existing beliefs), Revision (to add a new belief with possible removal of existing beliefs to result in a consistent belief state), and Contraction (to remove a set of beliefs). Since information should not be lost during the process of revision a set of minimal changes should be involved.

Theory (metamodel) definition and assumptions (instance)

The construction of a belief set is just part of the process of incrementally creating a consistent theory. The natural language processing and the knowledge model actual creation are the core tasks that need to be automated so the revision task can occur.

The actual processing of text and model creation is detailed in Section 2.

Nouns, modifiers, and verbs

- Text processing for metamodel definitions
- Text processing for model instantiation based on metamodel definition

2 Natural Language Processing

Input ———

A text represents a theory that need to be interpreted so it can be useful in the problems solving tasks.

A natural language parser formed by a part-of-speech (POS) tagger and a phrase structure parser are used to process the input text and add linguistic meta-information about the text. This text with attached linguistic information will be called tagged text.

Once the input text is tagged it can be processed by a context-free grammar parser. In our case we use model-based frameworks to process the tagged text and create a model-based representation of the text. The Textual Concrete Syntax (TCS) tool defines the tagged text syntax and a outputs the tagged text as a Ecore model. The Kernel meta meta model (KM3) tool is used to specify the tagged text meta model.

Goals ———

Like Simplified English (SE): " SE consists of an extensive series of rules and restrictions ranging from permissible grammatical structures to sentence and paragraph length. Its most important feature is probably the prescribed base vocabulary of about a thousand carefully selected words. Generally, there is only one word for a given concept and a word can only be used in one way. The word spring, for example, is acceptable only as a noun, not as a verb. Aerospace manufacturers are allowed to augment this general vocabulary with designated nouns and verbs for specific technical names and processes. Boeing has defined a company-specific technical vocabulary of about 2700 words for SE. "

" The Boeing Checker, which was based on work done for a natural language understanding project and had a gestation period of just one year, is built around a syntactic analyzer containing a tokenizer, a lexicon, a parser, and grammar containing more than 350 syntactic rules for English. It is sensitive to the distinction made in the documentation between procedural text, descriptive text, and notes. In procedural text, for example, only the active voice is allowed and the maximum sentence length is twenty words. Descriptive text has slightly relaxed requirements in these respects, allowing for sentences up to twenty-five words of length and no more than one passive per six sentences. "

one of the goals of this project is to make the written document easier to understand. This approach does not only improve the documentation interpretation by humans but simplifies the task of processing such text by a computer program.

The adequate understanding documents shared by the community of stakeholders, users, and developers need to be promoted. A common semantic, an agreement, along the used terms (words) and actions (relations) is necessary as a starting point to develop a framework (theory) that uses the shared concepts.

Text Pattern Validation ———

Before the tagged text can be used for any particular purpose, such as software specification, it is analyzed and validate so if any not expected phrasal structure exist it can be used to extend the tagged text parser, which is implemented in TCS.

The tagged text should follow a set of standards so the intended meaning of the phrases can be well understood by readers, avoiding misunderstandings.

A list of the desired patterns are:

- 1) The text that specifies
- 2)

Text Reasoning ———

Once the whole tagged text is validated, that means, all the senteces are written according to the previously specified grammatical structure patterns, it is possible to reason about the definitions expresses in the natural language written specification.

The initial reasoning about the text focus on the derivation of a Ontology of definitions and relations.

3 Reasoning Rules

Each word in a text is tagged according to the POS classification. This meta-information is used to identify the phrasal structures in sentences. Despite of POS tags being important, it is only possible to identify entities and possible relations, such as specialization, in the phrasal structure context.

The following sections present the two phases necessary to accomplish reasoning over any theory that is specified through natural language.

3.1 Theory (metamodel) definition

A initial set of beliefs are documented and stored separately from the actual text that contain analysis over the constructed theory. These beliefs shall define a metamodel that conformes to a core metamodel, which is called metametamodel. This core model represent basic elements essential for the definition of any metamodel. The theory definition is validated through the same rules that validate any metamodel that conforms to the core model.

The input text does not only contain metamodel definitions, there are special rules that shall be transtaled into validation rules that can be applied to the model instance.

Nouns and modifiers In a sentence, when nouns are KM3 definitions, adjectives are seen as new types that will form the new metamodel. When nouns are not KM3 definitions they represent references to new classes. In this case any existent modifier represents its new sub-type.

KM3 definitions should not be used as noun modifiers, if so, a problem should be reported. Analyzing the sentence: "A peer is any networked entity that implements one or more of the JXTA protocols", we have that:

1. the word "peer" is not a KM3 definition, therefore it is a new class. Because it is not modified no sub-class is being specified.
2. The term "entity" is also a new class and "networked" is a possible sub-type.
3. The same goes for "protocols", as a new class, and to "JXTA", as its sub-class.

Verbs In a sentence, when the used verbs are KM3 verbs (a set of verbs with predefined meaning that refer to KM3 relations) the phrase elements (types) are related according to the verb meaning (is/are (to be), to have, to contain, etc). When the used verb is not a KM3 verb, a problem should be reported.

If the verb is not a KM3 verb, it will define a new class that have as attribute references to the subject and the object types. Analyzing the sentence: "A peer is any networked entity that implements one or more of the JXTA protocols", we have that:

1. the verb "is" refers to the extension relation in KM3, therefore the subject type should extend the object type. In this case the object type, which is probably a noun, should be treated as specified before.
2. The verb "implements" is not a KM3 verb, it means the Implements class is created and "entity" and "[one or more] JXTA protocols" are added as attributes. There are a set of predefined tokens that represent KM3 relations.

3.2 Text processing for model instantiation based on metamodel definition

Nouns and modifiers When nouns are metamodel or KM3 definitions, adjectives are seen as instances of the types. This instance can be validated based on the model constraints. These instances are assumptions about the systems, not facts about the system's domain terminology (theory).

If nouns are neither metamodel nor KM3 definitions a problem should be reported since the model instance should only refer to instance of metamodel elements. In this case any existent modifier represents also represents a problem since it is specifying the instance of a unknown type.

Analyzing the sentence: "A peer is any networked entity that implements one or more of the JXTA protocols", we have that:

1. The "peer" nouns refers to a new instance of a networked entity.
2. A "networked entity" should be a class of the referred metamodel, OR
3. If "networked" or "entity" are not valid types in the metamodel, then a problem should be reported.

Verbs If the used verbs is a KM3 verb, the a model instance is created according to the verb meaning; if the subject and object of the phrase are not related according to the metamodel definition a problem should be reported.

When the used verb belong to the metamodel, a instance of the class should be created with the respective attribute values. If wrong attributes are given (same problem as described in the last paragraph), a problem is reported.

Analyzing the sentence: "A peer is any networked entity that implements one or more of the JXTA protocols", we have that:

1. The verb "is" refers to the extension relation in KM3, therefore the subject type should extent the object type. In this case the verb's object type ("networked entity") should be a metamodel element and the subject the instance.
 2. The verb "implements" is not a KM3 verb, it means the Implements class is created and "entity" and "[one or more] JXTA protocols" are added as attributes. There are a set of predefined tokens that represent KM3 relations. Because it is not a KM3 verb the "implement" class is only created if it was specified in the metamodel definition.
-

3.3 Text processing notes

The way a sentence is constructed (tense, coordination, subordination, etc) reveals different meanings, each case is treated separately and

4 Final Considerations