RC23372 (W0410-073) October 12, 2004
Computer Science

IBM Research Report

Text Analysis as Formal Inference for the Purposes of Uniform Tracing and Explanation Generation

David Ferrucci
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

IBM T.J. Watson Research Center

Text Analysis as Formal Inference for the purposes of Uniform Tracing and Explanation Generation

By David A. Ferrucci

1 The KANI Objective

The high-level research goal of the Knowledge Associates for Novel Intelligence (KANI) project is to use a variety of advanced techniques to transform unstructured information, in the form of natural language text documents, into actionable knowledge. That is, knowledge that is sufficiently structured and assigned precise semantics so that classic automating reasoning techniques maybe applied to generate and test intelligence analyst's hypotheses. The project brings together experts in text extraction, knowledge representation and reasoning, and advanced user interfaces from Stanford, IBM Research, and Battelle.

The practical realization of this goal is an interactive system capable of assisting the intelligence analyst in quickly discovering, extracting, filtering and synthesizing complex unstructured data to arrive at a manageable task-relevant working knowledge-base (WKB). Through the application of automated reasoning techniques, KANI will assist the user in generating and testing hypotheses over the assertions and rules in the working knowledge-base.

A key element of KANI's interaction with the user is the ability of KANI to *explain* its final and intermediate inferences. KANI must leave open the possibility that any inference made in the process of supporting or refuting a hypothesis may be brought under question by the analyst. To do this, KANI must provide a formal mechanism for tracing its reasoning from the top-most conclusions back through the lowest-level inferences.

This report proposes a key innovation for UIMA to explain text analysis as a series of high-level inference steps.

2 Background

The KANI project has adapted, extended, and reused several existing technologies. Most relevant to this paper are the Inference Web and UIMA.

2.1 The Inference Web

The Inference Web (IW) is a technology for generally representing and tracing the operation of formal inference systems. The Inference Web technology therefore provides a general representational and technical infrastructure for producing explanations for automatically generated inferences that is independent of the particular reasoning engine or imposed logic [3,4,5].

The Inference Web is the principal technology for producing explanations for inferences generated by the KSL reasoning component designed for KANI, the The Devil's Advocate. This is a natural application of the IW technology since the Devil's Advocate uses a first-order theorem prover with a sound and complete declarative semantics [6]. The inferences the theorem prover can make, such as modus ponens, are registered with the inference web as part of the semantics of the language used.

2.2 UIMA: An Architecture for Text Analytics

UIMA is IBM's architecture for describing, implementing, composing and deploying multi-modal analysis algorithms, such as those that process natural language text [1, 2].

IBM's role on the KANI project is to produce processes that discover semantic entities and relationships in natural language text documents, extract their linguistic representations and map them to formal knowledge representations.

Text analysis processes are typically not encoded by their developers in terms of declarative axioms and rules. Rather they are encoded in procedural form using a wide variety of algorithm techniques ranging from linguistic grammars to statistically trained machine learning algorithms. Their function is typically to assign semantics to regions of text by associating one or more of an established set of annotations with different regions in a document. We call this annotating the text or producing annotations.

For example a text analysis process may determine that the word "Bush" in a sentence refers to a person name, and it may further determine that it refers to the same person that the name "Present George W. Bush" refers to in another document.

UIMA is a software architecture and framework that supports the encapsulation, declarative description and hierarchical composition of these analysis algorithms. It also provides tooling for aiding in the construction of complex analysis processes from more primitive ones.

3 The Problem

A key requirement in KANI is that all inferences made in the process of transforming unstructured information to actionable knowledge be accessible to the user in the form of explanations.

Inferences by the Devil's Advocate, a formal reasoning system with declarative semantics, are naturally managed by the Inference Web. The problem is that text analysis processes are making important contributions to the actionable knowledge, however their results are not cast as formal inferences, neither in their encodings, nor in their output, and as a result they do not map naturally to the IW formalism.

From the user's perspective, the text analysis processes are making just as important inferences as the Devil's Advocate, and furthermore the results are logically linked. The text analysis processes are interpreting expressions in raw text, assigning semantics and forming many of logical assertions that make up the WKB that are used by The Devil's Advocate. Tracing backwards, for example from a top-level conclusion reasoned by Devil's Advocate, the explanation facility can not see beyond the Devil Advocate's rationale, in spite of the fact that many of its primary assertions were determined by the text analysis processes.

KANI needs to produce a uniform view and explanation facility that spans the formal, logic-based inferencing approach employed by the Devil's Advocate and the highly-varied and procedural approaches employed by the text analysis process.

4 A Solution Approach

A solution to this problem would allow the Inference Web to uniformly treat the text analysis processes as it would any other formal inference system.

This solution would require that the text analysis processes are somehow mapped to formal declarative inferences using the Inference Web's portable proof mark-up language.

While text analysis process are ultimately encoded using a host of different procedural algorithms, UIMA does impose some structure on their encapsulation, composition and behavioral description.

Key features of UIMA that would enable this mapping are:

1) Algorithms are encapsulated in modules, called Analysis Engines, whose behavior, in terms of their input and output, must be declaratively represented in terms of a common Type System. A type system is a simple ontology.

2) All Analysis Engines have a common interface and may be structurally composed by other analysis engines. This hierarchical structure leads to a natural and discoverable decomposition of function.

With these two features we can imagine UIMA analysis engines mapping to inference steps and the annotation of text mapping to a type of logical reasoning. These inference steps may be unsound, or incomplete, but the IW is not a reasoning system itself, rather a framework for *tracing and explaining* inference processes, and is not limited to sound and complete reasoning systems.

The innovation here is to take a high-level, abstract view of text analysis components in which the components operate by applying *rules*. At the detailed level these components are purely procedural, and may use lexical or syntactic rules to analyze the text, but at an abstract level they take text and other annotations as input and produce new annotations as output. The high-level "inference steps" they make may be, for example, to conclude that a span of text is a Person's name, or that a Person and a Place annotation produced a "located-in" annotation. Note that the proposal is not to explain in much detail how the annotators concluded this, just enough to record the data flow in a way that a person could understand what happened.

Tracing through a text-analysis process by drilling down through the compositional structure of UIMA analysis engines from their conclusions all the way to the raw text expressions, would be analogous to producing a proof that drills down from conclusions of rules through their premises down to ground facts.

What is needed to make this solution approach effective is a classification of the annotations UIMA analysis engine can produce as type of declarative inference and their formalization in the Inference Web's proof mark-up language.

5 Acknowledgements

This work performed for the KANI projects and partially funded by ARDA under the Novel Intelligence from Massive Data (NIMD) program. Contract #: 2003*H278000*002-2.

IBM is one of two subcontractors to Stanford's Knowledge Systems Laboratory (KSL). The other is Battelle. The three groups are working together to address a single research objective in the development of the KANI system.

6 References

[1] D. Ferrucci and A. Lally, "UIMA by Example," *IBM Systems Journal* **43**, No. 3, 455-475 (2004).

[2] D. Ferrucci and A. Lally, A. " UIMA: an architectural approach to unstructured information processing in the corporate research environment," *Natural Language Engineering* **10**, No. 3-4, 327-348 (2004)

[3] Pinheiro da Silva, P., Hayes, P. J., McGuinness, D. L., & Fikes, R. 2004. PPDR: A Proof Protocol for Deductive Reasoning. Technical Report, Knowledge Systems Laboratory, Stanford University

[4] Pinheiro da Silva, P., McGuinness, D. L., & Fikes, R. 2004. A Proof Markup Language for Semantic Web Services. *Information Systems Journal* (to appear)

[5] Pinheiro da Silva, P. McGuinness, D. L., & McCool, R. 2003. Knowledge Provenance Infrastructure. *IEEE Data Engineering Bulletin*, 26(4), pages 26-32

[6] Fikes, Richard, Jessica Jenkins, and Gleb Frank. "JTP: A System Architecture and Component Library for Hybrid Reasoning." *Proceedings of the Seventh World Multiconference on Systemics, Cybernetics, and Informatics*. Orlando, Florida, USA. July 27 - 30, 2003.