U-Compare Type System

1 BASIC INFORMATION

1.1 Resource composition

The resource constitutes of a hierarchically-structured system of data types, which is intended to be suitable for describing the inputs and output annotation types of a wide range of natural language processing applications which operate within the UIMA Framework¹ (Ferrucci et al, 2006). It is being developed in conjunction with the U-Compare Workbench, but can be used as the base type system for other UIMA components and workflows, to help to ensure greater interoperability.

1.2 Representation of the resource (flat files, database, markup)

The resource is provided as java archive (jar file), UCompareTypeSystem.jar

1.3 Character encoding

The characters are UTF8 encoded.

2 ADMINISTRATIVE INFORMATION

2.1 Contact person

Name: Sophia Ananiadou Address: Manchester Interdisciplinary Biocentre,131 Princess Street, Manchester M1 7DN, IK Affiliation: National Centre for Text Mining, School of Computer Science, University of Manchester Position: Director Telephone: +44 161 306 3092 Fax: +44 161 306 5201 e-mail: Sophia.Ananiadou@manchester.ac.uk

2.2 Delivery medium (if relevant; description of the content of each piece of medium)

The resource is available on the MetaShare platform.

2.3 Copyright statement and information on IPR

The U-Compare workbench is released under a dual license, the LGPL open source license (<u>http://www.gnu.org/licenses/lgpl.html</u>) or a

¹ http://uima.apache.org/

commercial license. Please use the contact details above if you are interested in obtaining a commercial licence.

3 TECHNICAL INFORMATION

3.1 Directories and files

The jar file contains XML descriptor files corresponding to the core U-Compare type system, as well as various extensions that have been created to accommodate various components in the U-Compare library. Also included in the archive are automatically generated Java source and class files, corresponding to each annotation type in the system. These are required by the UIMA framework, since each annotation in the Common Analysis Structure (CAS; the common data structure that is used to store the results produced by each component in a UIMA workflow) corresponds to a Java object.

3.2 Data structure of an entry

The file follows the required format of UIMA type system descriptor files², which are XML files. The following is an example of the description of an individual type.

```
<typeDescription>
  <name>org.u compare.shared.syntactic.POSToken</name>
  <description/>
   <supertypeName>org.u_compare.shared.syntactic.Toke
 n</supertypeName>
    <features>
       <featureDescription>
              <name>pos</name>
                <description/>
                <rangeTypeName>
                   org.u_compare.shared.label.POS
               </rangeTypeName>
       </featureDescription>
       <featureDescription>
                 <name>posString</name>
                 <description>A special field which
 internally converts object of pos field into String
 value.
                 </description>
                 <rangeTypeName>uima.cas.String
                 </rangeTypeName>
      </featureDescription>
    </features>
</typeDescription>
```

² See <u>http://uima.apache.org/downloads/releaseDocs/2.1.0-</u> incubating/docs/html/tutorials_and_users_guides/tutorials_and_users_guides.ht ml#ugr.tug.aae.defining_types

The tags and attributes are as follows:

- typeDescription contains the description of the type
 - o *name* the name (label) assigned to the annotation
 - *description* a textual description of the type
 - *supertypeName* types in the type system are hierarchically structured. This element contains the name assigned to the supertype of the current type.
 - *Features* contains descriptions of the features (attributes) associated with the current annotation type.
 - *featureDescription* Contains the description of a single feature/attribure
 - *name* the name of the feaure
 - *description* a textual description of the feature
 - *rangeType* the type of the the value of the feature

3.3 Resource size (nmb. of tokens, MB occupied on disk)

The U-Compare type system consists of a hierarchy of 281 different types. The size of the file is 89 KB.

4 CONTENT INFORMATION

4.1 The natural language(s) of the resource

The type system was developed mainy based on the inputs/ouitputs of English tools, although recent work on developing UIMA components for other languages suggests that the current type system is largely suitable for other languages, at least European ones.

4. 2 Entry Type

The data types described by the U-Compare type system can be roughtly split into three different group, i.e., syntactic types, semantic types and document types (i.e, describing the structural aspects of a document).

4.3 Attributes and their values

As mentioned in the section 3.2, each type in the type system may have zero or more attributes associated it, to store additional information about the annotations.

4.4 Coverage of the resource

Currently, the U-Compare type system is only suitable for text-based UIMA components, although extensions are planned for speech-based applications. Figures 1, 2 and 3 provide an overview of the main data

types covered by the U-Compare type system, and the hierarchical structure of these types.

4.5 Intended application of the resource

In UIMA, a common data structure called the Common Analysis Structure (CAS) is used to store the outputs of each component in a workflow, in the form of annotations. Each component obtains its input by reading relevant annotations from the CAS, and produces output by creating nre annotations in the CAS, or updating existing annotations. The UIMA framework itself does not attempt to place any restrictions or recommendations regarding the use of a particular system of annotation types. However, some level of commonality of the type systems used by different components is required to try to achieve maximum interoperability and flexibility in the ways in which different components can be combined. For example, if a named entity recogniser requires the input types *Token* and *Chunk*, then it is only possible to use components that produce annotations with these names earlier in the workflow.

Different NLP research groups have produced different repositories of UIMA components, e.g., the BIONLP UIMA Component Repository (Baumgartner et al., 2008), the CMU UIMA component repository³ and the UIMA-fr consortium (Hernandez et al., 2010), but generally using their own type systems. This can cause problems for interoperability - components developed by one team cannot be combined easily with components developed by another team, because they use different type systems.

Ideally, to achieve maximum interoperability, a single, common type system would be imposed, to be followed by all developers of NLP UIMA components. However, this is considered not a viable option, as it would be difficult to achieve consensus on exactly which types should be present, given, for example, the various different syntactic and semantic theories on which different tools are based.

The U-Compare type system is a *sharable* type system, which aims to cover the most common types of annotation, both syntactic and semantic, that are produced by NLP applications. The idea is that all components in a UIMA workflow should produce annotations that are compatible with this type system. As the U-Compare type system consists of fairly general types, it is permissible to create new types that correspond to more specialised types of annotations, as long as these new types can form sub-types of one of the existing U-Compare types. This ensures that compatibility between components developed by different groups can at least be achieved at an intermediate level of the hierarchy.

³ <u>http://uima.lti.cs.cmu.edu</u>



Figure 1: Main syntactic types in the U-Compare type system



Figure 2: Main semantic types in the U-Compare type system



Figure 3: Main document-level types in the U-Compare type system

4.6 Reliability (automatically/manually constructed)

The U-Compare type system has been manually constructed, by considering the different input and output types of a wide range of NLP and text mining tools, and is still evolving. The success of the U-Compare type system in facilitating the construction of interoperable UIMA components can be demonstrated by the fact around 60 components that comply with the U-Compare type system, and which cover a number of different European languages, are now available in the component library of the U-Compare Workbench. This library is being extended as part of the META-NET initiative (Ananiadou et al., 2011; Thompson et al. 2011)

5 RELEVANT REFERENCES AND OTHER INFORMATION

Ananiadou, S., Thompson, P., Kano, Y., McNaught, J., Attwood, T. K., Day, P. J. R., Keane, J., Jackson, D. and Pettifer, S. (2011). "Towards Interoperability of European Language Resources". *Ariadne*, 67

Baumgartner, W. A., Cohen, K. B., & Hunter, L. (2008). "An open-source framework for large-scale, flexible evaluation of biomedical text mining systems". *Journal of Biomedical Discovery and Collaboration, 3*, 1.

Ferrucci, D., Lally, A., Gruhl, D., Epstein, E., Schor, M., Murdock, J. W., Frenkiel, A., Brown, E.W., Hampp T., Doganata, Y., Welty, C., Amini, L., Kofman, G., Kozakov, L. and Mass, Y. (2006). Towards an Interoperability Standard for Text and Multi-Modal Analytics. IBM Research Report RC24122.

Hernandez, N., Poulard, F., Vernier, M., & Rocheteau, J. (2010). "Building a French-speaking community around UIMA, gathering research, education and industrial partners, mainly in Natural Language Processing and Speech

Recognizing domains". In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp 41-45

Kano, Y., Baumgartner, W. A., Jr., McCrohon, L., Ananiadou, S., Cohen, K. B., Hunter, L. (2009). "U-Compare: share and compare text mining tools with UIMA". *Bioinformatics*, vol. 25, no. 15, 1997-1998.

Kano, Y., Miwa, M., Cohen, K. B., Hunter, L. E., Ananiadou, S., & Tsujii, J. (2011). "U-Compare: A modular NLP workflow construction and evaluation system". *IBM Journal of Research and Development*, *55*(3), 11:11-11:10.

Thompson, P., Kano, Y., McNaught, J., Pettifer, S., Attwood, T. K., Keane, J. and Ananiadou, S. (2011). Promoting Interoperability of Resources in META-SHARE. In *Proceedings of the IJCNLP Workshop on Language Resources, Technology and Services in the Sharing Paradigm (LRTS)*, pp. 50-58