

Using the MetaMap UIMA Annotator

Willie Rogers

December 16, 2013

1 Purpose

MetaMap maps terms occurring in text to UMLS Metathesaurus concepts. As part of this mapping process, MetaMap tokenizes text into sections, sentences, phrases, terms, and words. MetaMap maps the noun phrases of the text to the best matching UMLS concept or set of concepts that best cover each phrase. The MetaMap Java API provides java programs with programmatic access to MetaMap mapping engine. Additionally, this annotator encodes MetaMap named entities in a format utilizable by UIMA components. The annotator is based on the MetaMap UIMA Wrapper (<http://sourceforge.net/projects/metamap-uima/>) authored by Kai Schlamp.

2 Audience

This document assumes that the user has adequate knowledge of Java software development and the Apache UIMA Framework in particular. Knowledge pertaining to using the UIMA framework with the Eclipse Integrated Development Environment is useful, but not required.

3 Pre-requisites

The full MetaMap download and installation is required to use the MetaMap UIMA Annotator (see <http://metamap.nlm.nih.gov/#Downloads>). Also, Java 1.6 SDK or greater is required. The UIMA Annotator also uses classes from MetaMap Java API (<http://metamap.nlm.nih.gov/#MetaMapJavaApi>), so you'll need to install that also.

4 Extracting and Installing the API distribution

After downloading the MetaMap UIMA Annotator archive, in the directory where you extracted the Public Metamap (the directory containing the public_mm directory) extract the uima archive (Note: the annotator archive must be extracted AFTER the main, and api archives):

```
$ bzip2 -dc /home/piro/public_mm_uima_2012.tar.bz2 | tar xvf -
```

or on Windows 7 extract the 7-Zip or zip distributions using an Archive tool such as WinZip (www.winzip.com) or 7-Zip (<http://www.7-zip.org/>).

You will need to run ./bin/install.sh from the public_mm directory to setup the files for MetaMap, the Java API, and the UIMA Annotator.

```
$ ./bin/install.sh
```

or on Windows click on the “Install MetaMap” icon in the public_mm folder.

Important: be sure to respond to any prompts whose defaults do not match your system’s configuration.

5 Testing UIMA annotator using documentAnalyzer

To test the annotator installation, from the public_mm directory first run the MedPost Tagger Server, the WSD servers (if necessary, it’s optional), and the MetaMap server (mmserver10). The MedPost Tagger Server and the WSD Server will run in the background automatically; the MetaMap server, however, runs as a foreground process.

```
$ bin/skrmedpostctl start
Starting skrmedpostctl:
started.
$ bin/wsdserverctl start
Starting wsdserverctl:
started.
loading properties file /Users/dotmatrix/public_mm/WSD_Server/config/disambServer.cfg
$ bin/mmserver10
/Users/dotmatrix/public_mm/bin/SKRrun
-L 2010 -w /Users/dotmatrix/public_mm/lexicon
/Users/dotmatrix/public_mm/bin/mmserver10.BINARY.Linux -Z 10
Server options: [port(8066),accepted_hosts(['127.0.0.1'])]
Berkeley DB databases (normal 10 strict model) are open.
Static variants will come from table varsan in
/Users/dotmatrix/public_mm/DB/DB.normal.10.strict.
Derivational Variants: Adj/noun ONLY.
Accessing lexicon /Users/dotmatrix/public_mm/lexicon/data/BDB4/lexiconStatic2010.
Variant generation mode: static.
```

To run the annotator using the UIMA environment, source the UIMA setup script and then run the UIMA document analyzer:

```
$ source bin/setup_uima.sh
$ documentAnalyzer.sh
```

See the document “Getting Started: Installing the Java UIMA Framework and SDK, and Running Examples” (<http://uima.apache.org/doc-uima-examples.html>) for more information on using the UIMA document analyzer.

6 The Annotator Sources

The source code for the annotator and build scripts are located in the public_mm/src/uima directory. The source to the primary class, MetaMapAnnotator is in the subdirectory src in the

package subtree `gov/nih/nlm/nls/metamap/uima`. The automatically generated sources for the UIMA type system used by the Annotator reside in the subdirectory `ts_src` in the package subtrees `gov/nih/nlm/nls/metamap/uima/ts` and `org//metamap/uima/ts` (modified versions of Kai Schlamp's MetaMap UIMA wrapper type system classes).

7 Modification of the Type System

If necessary, the Eclipse IDE and UIMA plugins provide the most expedient means for modifying the UIMA type system used for the annotator. It is also possible to modify the type system directly by editing type system descriptor files directly. See the "Apache UIMA Documentation" (http://uima.apache.org/documentation.html#manuals_and_guides) for information on using Eclipse and the UIMA plugins as well as the use of UIMA components.

8 Using the MetaMap UIMA annotator with Eclipse

8.1 Setup UIMA in Eclipse (Optional)

This is only necessary if you plan to change the Analysis Engine or TypeSystem used by Annotator. Please review Chapter 3. of "UIMA Overview & SDK Setup" manual: "Setting up the Eclipse IDE to work with UIMA" (http://uima.apache.org/d/uimaj-2.4.0/overview_and_setup.html#ugr.ovv.eclipse_setup) before continuing.

Note: the Apache UIMA Eclipse plugin site now is at the url <http://www.apache.org/dist/uima/eclipse-update-site/> (this is subject to change, check with the uima.apache.org site for the latest information).

8.2 Creating a MetaMap UIMA Project in Eclipse

Create a new "Java Project" by selecting from the "File" menu -> "New" -> "Java project".

Fill in the name of your project in the "Project name:" entry box.

Select "Next" button to go to "Java Setting Panel".

In the "Libraries" tab add your UIMA libraries using the "Add External JARs" button.

These necessary libraries include the following:

```
jVinci.jar
uima-adapter-soap.jar
uima-adapter-vinci.jar
uima-core.jar
uima-cpe.jar
uima-document-annotation.jar
uima-examples.jar
uimaj-bootstrap.jar
uima-tools.jar
```

Add the MetaMap Java API jar archive at `public_mm/src/javaapi/dist/MetaMapApi.jar`.

Add the PrologBeans jar file at `public_mm/src/javaapi/dist/prologbeans.jar`.

Add the MetaMap UIMA jar archive at `public_mm/src/uima/lib/metamap-api-uima.jar`.

Add the MetaMap UIMA descriptor directory at `public_mm/src/uima/desc`.

Click the “Finish” button at the bottom of the panel to create the project.

8.3 Running the UIMA Document Analyzer Application in Eclipse

Select “Project” menu -> “Properties” to get a “Properties ...” panel for your application.

Select “Run/Debug” from the list on the left.

Select the “New” Button of right. Select “Java Application” in the Pop-up panel.

In the “Edit Configuration” panel name the app “documentAnalyzer”.

Select “Search” button at right of “Main class:” entry box.

Select “documentAnalyzer” from “Select Main Type” panel and click “OK”.

Click “OK” in “Edit Configuration”.

Click “OK” in “Properties ...” Panel.

Select your Application in the “Package Explorer” and then select “Run” menu -> “Run Configuration”. You should see “DocumentAnalyzer” in the left list. Select it.

Make sure both the Tagger Server (`skrmedpostctl`) and the Metamap server (`mmserver`) are running. (if you plan to use Word Sense Disambiguation make sure `wdsverctl` is running as well.)

Select the “Run” button at the bottom of the “Run Configurations” panel. The “documentAnalyzer” program should appear.

Add the MetaMap API Analysis Engine XML Descriptor (`public_mm/src/uima/desc/MetaMapAPIAE.xml`) to “Location of Analysis Engine XML Descriptor:” entry box.

To run a simple test select the “Interactive” button.

In the “Annotation Input” panel, insert the text “No increase in tumor size” (actually any medical oriented text will do...) and then press the “Analyze” button.

An “Annotation Results ...” panel should appear with results of the computation available in collapsable tree form.

9 Special Thanks to Kai Schlamp

Special thanks to Kai Schlamp and his MetaMap UIMA Wrapper (<http://sourceforge.net/projects/metamap-uima/>) on which many of the components of this project is based.