U-Compare Syntactic Parsing Service

1. BASIC INFORMATION

Service name

U-Compare Syntactic Parsing Service

Overview and purpose of the tool

This is a web service that carries out syntactic parsing on English plain text. Also identifies sentences and tokens, and assigns part-of-speech tags to tokens.

A short description of the algorithm

This web service is based on a UIMA-based workflow, created using the U-Compare text mining system[1]. The workflow was exported from U-Compare as a web service using the built in functionality (Kontonastsios et al., In Press). The workflow was created as part of the work to increase the number of interoperable tools operating on different European languages (Ananiadou et al, 2011).

The workflow consists of the following UIMA-compliant tools.

1) Cafetiere Sentence Splitter (University of Manchester)
2) OpenNLP Tokenizer (Apache)[2]
3) STEPP Tagger (University of Manchester) (Tsuruoka & Tsujii, 2005),
4) Enju Parser (University of Manchester) (Miyao & Tsujii, 2008)

2. TECHNICAL INFORMATION

Software dependencies and system requirements

This is a web service that can be run from a browser or accessed programmatically. The only basic requirement is an internet connection.

Installation

There is no installation. The web service can be accessed at the following URL:
http://nactem001.mib.man.ac.uk:8080/UCompareWebServices/SyntacticParsing_Enju

[1] http://nactem.ac.uk/ucompare/

[2] http://opennlp.apache.org/

The web form available at this URL is shown in Figure 1, with some English text entered into the text box.

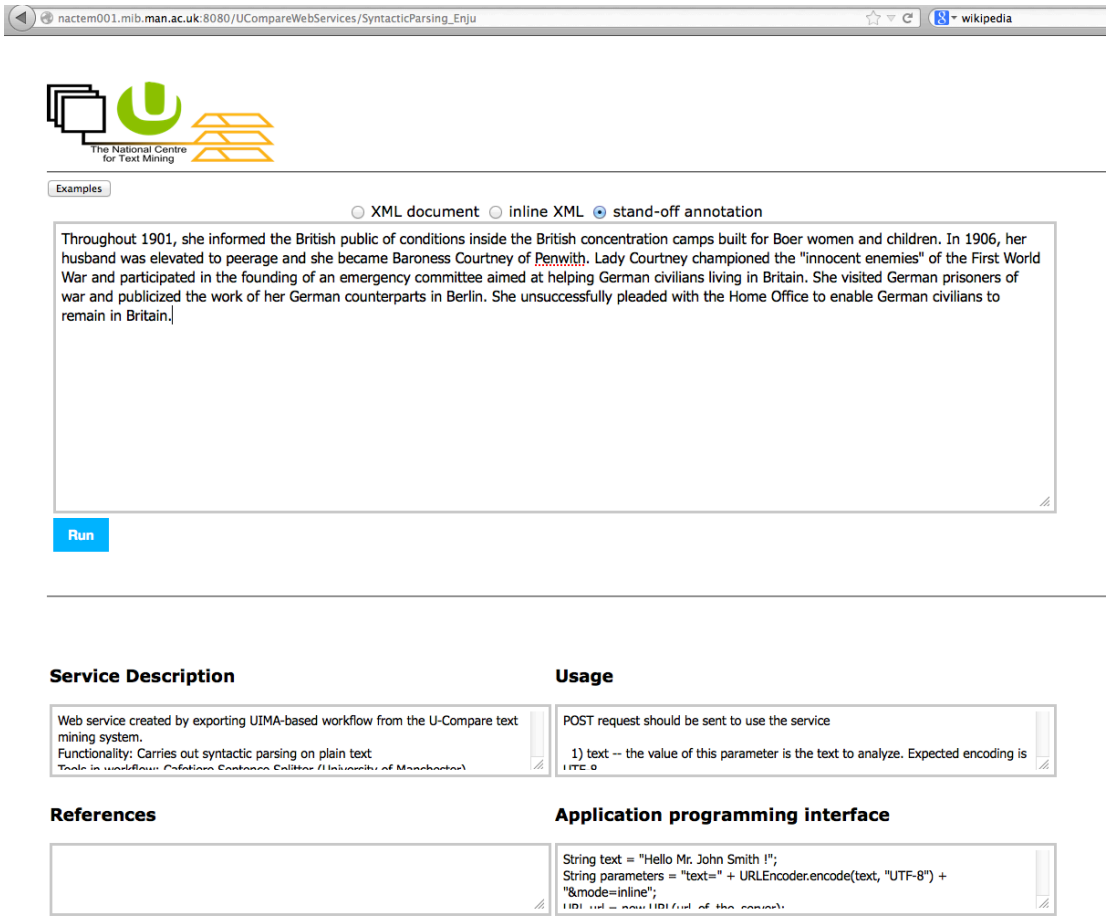

Figure 1: Web form for the web service

Execution instructions

The web service can be executed by typing or pasting text into the online form and clicking on the “Run” button.

Alternatively, the web service can executed from within program code, as explained in the “Usage” and “Application programming interface” boxes of the web form.

A POST request should be used to call the service. The following parameters may be used in the request:

- **text** - the value of this parameter is the text to analyze. Expected encoding is UTF-8. This parameter is obligatory.
- **lang** - This parameter sets the language of the text. If this parameter is not provided, then the value"en" will be used

- **mode** - This parameter sets the format of the annotated information returned by the service. If this parameter is not set, XML output will be produced. The two possible types of output are as follows:
    - **inline** – annotations are encoded as inline XML.
    - **xml** – results are output as an XML document containing the annotations added

The following code example shows how the web service can be called from Java code:

```
//Set the input text
String text = " <Text_to_be_analysed>";
//Set the parameter string
String parameters = "text=" + URLEncoder.encode(text,
"UTF-8") + "&mode=inline";
//Create the URL connection
URL url = new
URL("http://nactem001.mib.man.ac.uk:8080/UCompareWebServi
ces/SyntacticParsing_Enju");
URLConnection connection = url.openConnection();
connection.setDoOutput(true);
//Create Output stream
OutputStreamWriter writer = new
OutputStreamWriter(connection.getOutputStream());
//write parameters to output stream
writer.write(parameters);
writer.flush();

//Read the results returned by the service
BufferedReader reader = new BufferedReader(new
InputStreamReader(connection.getInputStream(), "UTF-8"));
String line;
while ((line = reader.readLine()) != null) {
    System.out.println(line);
```

}

Input/Output data formats

Input data formats

The input is plain text, UTF-encoded.

Output data format

If the service is run from the web interface, then the output is visualized in the interface using colored highlights in the text to show the individual annotations, and one or more tables of information below, each corresponding to a particular type of annotation.

If the service is run programmatically, then the output is provided in XML format. See section 3 for an example.

Integration with external tools

The API allows the functionality of the web service to be embedded in any application.

3. CONTENT INFORMATION

Using the web interface, the output of the service is visualised as shown in Figure 2.

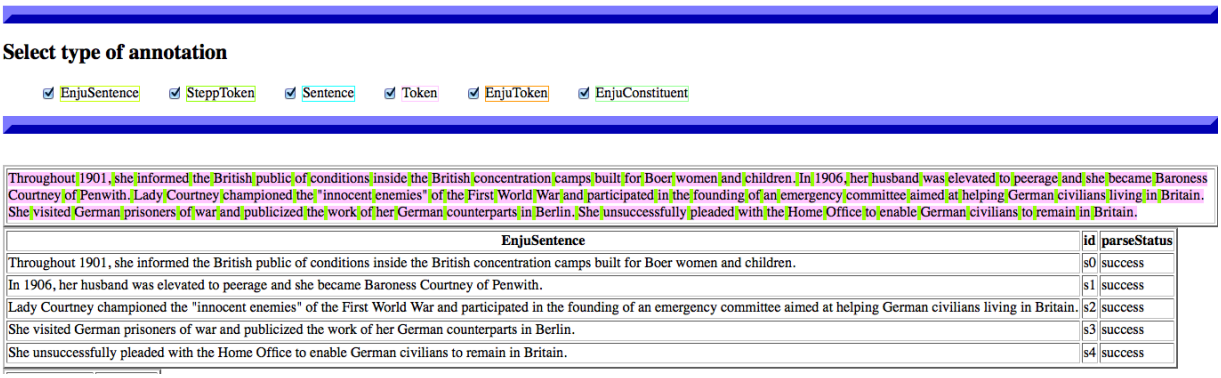

Figure 2: Visualisation of web service output

In Figure 2, the top of the screen has check boxes corresponding to each type of annotation produced by the workflow. Checking one or more of the boxes will cause the annotations to become highlighed in the view of the text below in different colours.

Below the text, the different types of annotations added by the workflow are shown in tabular format, with each type of annotation in a separate table. In Figure 2, a table of individual sentence annotations is shown.

In Figure 3, a further table of information is shown, corresponding to the different constituents identified in the text. Each is assigned a category ("cat") and id. The ID and the head and "semHead" of the constituent are also assigned, according to the HPSG formalism.

EnjuConstituent	id	headId	semHeadId	cat	xcat	schema
Throughout 1901, she informed the British public of conditions inside the British concentration camps built for Boer women and children.	c0	c5	c5	S		mod_head
Throughout 1901,	c1	c2	c2	PP		head_comp
Throughout	c2	t0	t0	PX		
1901,	c3	c4	c4	NP		empty_spec_head
she informed the British public of conditions inside the British concentration camps built for Boer women and children.	c5	c7	c7	S		subj_head
she	c6	t2	t2	NP		
informed the British public of conditions inside the British concentration camps built for Boer women and children.	c7	c8	c8	VP		head_comp
informed	c8	t3	t3	VX		
the British public of conditions inside the British concentration camps built for Boer women and children.	c9	c11	c11	NP		spec_head
the	c10	t4	t4	DP		
British public of conditions inside the British concentration camps built for Boer women and children.	c11	c12	c12	NX		head_mod
British public	c12	c14	c14	NX		mod_head
British	c13	t5	t5	ADJP		

Figure 3: Visualisation of Enju constituents

In Figure 4, information about individual tokens is shown. Links between tokens and constituents are made via ids are categorized (e.g., in the arg1Id and arg2Id columns). These links constitute the syntactic parse results.

EnjuToken	base	id	cat	lexEntry	pred	arg1Id	arg2Id			
Throughout	throughout	t0	P	[<P>NP.acc]V_lxm	prep_arg12	c5	c3			
1901,	-NUMBER-	t1	N	[D<N.3sg>]_lxm	noun_arg0					
she	she	t2	N	[<NP.3sg.nom>]	noun_arg0					
informed	inform	t3	V	past	active	minus	[NP.nom<V.bse>NP.acc]_lxm-past_verb_rule	none	verb_arg12	c6
the	the	t4	D	[<D>]N	det_arg1	c11				
British	british	t5	ADJ	[<ADJP>]N_lxm	adj_arg1	c14				
public	public	t6	N	[D<N.3sg>]_lxm	noun_arg0					
of	of	t7	P	N[<P>NP.acc]_lxm	prep_arg12	c12	c17			
conditions	condition	t8	N	[D<N.3sg>]_lxm-plural_noun_rule	noun_arg0					
inside	inside	t9	P	N[<P>NP.acc]_lxm	prep_arg12	c19	c22			
the	the	t10	D	[<D>]N	det_arg1	c24				

Figure 3: Visualisation of Enju tokens with links to constituents

annotations and one for "RichToken" annotations. For each annotation type, the information associated with each annotation in shown in a row of the table. For sentences, the information comprises only the text covered by the sentence annotation. For "RichToken" annotations, the information additionally includes the part-of-speech tag assigned to the token (in the "posString" column) and the lemma (in the "base" column).

An example of the XML output format, which is more suited to programmatic use, is shown in Figure 5. In the XML, each type of annotation is represented by a different XML element type, and the information associated about each token in encoded in the attributes of the XML element.

Figure 5: XML output example

3. LICENCE

a) The web service only is licenced NaCTeM Web Service Licence Agreement (standard non-commercial use) – see "U-Compare-Syntactic-Parsing-Service-Licence.pdf" in the "licences" directory. Please contact us using the details below if you require a commercial licence.

b) The tools used in the workflow on which the web service is based may have their own licences. The NaCTeM Web Service Licence Agreement does NOT apply to these tools.

4. ADMINISTRATIVE INFORMATION

Contact

For further information, please contact Sophia Ananiadou:
sophia.ananiadou@manchester.ac.uk

5. REFERENCES

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.

Kontonatsios, G., Korkontzelos, I., Kolluru, B., Thompson, P. and Ananiadou, S. (In Press). Deploying and Sharing U-Compare Workflows as Web Services. *Journal of Biomedical Semantics.*

Yusuke Miyao and Jun'ichi Tsujii. 2008. Feature Forest Models for Probabilistic HPSG Parsing. *Computational Linguistics*. 34(1):35--80, MIT Press

Yoshimasa Tsuruoka and Jun'ichi Tsujii. 2005. Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of HLT/EMNLP 2005*. pages 467–474.