TTC TermSuite
A UIMA Application for Multilingual Terminology Extraction from Comparable Corpora

Jérôme Rocheteau **Béatrice Daille**
University of Nantes - LINA
2, rue de la Houssinière
BP 92208 – F-44322 Nantes cedex 3
{jerome.rocheteau,beatrice.daille}@univ-nantes.fr

Abstract

This paper aims at presenting TTC TermSuite: a tool suite for multilingual terminology extraction from comparable corpora. This tool suite offers a user-friendly graphical interface for designing UIMA-based tool chains whose components (i) form a functional architecture, (ii) manage 7 languages of 5 different families, (iii) support standardized file formats, (iv) extract single- and multi- word terms languages by languages (v) and align them by pairs of languages.

1 Introduction

Lexicons and terminologies play a central role in any machine translation tool, regardless of the theoretical foundations upon which the machine translation (MT) tool is based (e.g. statistical machine translation or rule-based machine translation, example-based translation, etc.). Terminologies may be extracted from parallel corpora, i.e. from previously translated texts, but such corpora are scarce. Previously translated data is still sparse and only available for some pairs of languages and few specific domains, such as Europarl (Koehn, 2005). Thus, no parallel corpora are available for most specialized domains, especially for emerging domains. Several tool suites exist for multilingual term extraction for parallel corpora: the GIZA++ statistical machine translation toolkit (Och and Ney, 2003), the iTools suite that performs single- and multi- word alignment, and includes graphical and interactive tools (Merkel and Foo, 2007). To tackle the drawbacks of term alignment from parallel corpora, comparable corpora that are "sets of texts in different languages that are not translations of each other" (Bowker and Pearson, 2002, p. 93) seem to be the right solution to solve textual scarcity. The bilingual alignment is performed thanks to contextual analysis such as (Rapp, 1995). TTC TermSuite is the first tool suite for the multilingual extraction of terminology from comparable corpora. It is multilingually designed, adopting a 4-step functional architecture and using the UIMA open solution.

TTC TermSuite is designed to perform bilingual term extraction from comparable corpora in five European languages: English, French, German, Spanish and one under-resourced language, Latvian, as well as in Chinese and Russian. TTC TermSuite is a 4-step functional architecture that is driven by the required inputs and provided outputs of each tool. The bilingual term alignment (step 4) requires processes of monolingual term extraction (step 3), itself requiring preliminary linguistic analysis (step 2) that requires text processing (step 1). TTC TermSuite is based on the UIMA framework which supports applications that analyze large volumes of unstructured information. UIMA was developed initially by IBM (Ferrucci and Lally, 2004) but is now an Apache project[1]. UIMA enables such applications to be decomposed into components (and components into sub-components) and to aggregate the latter easily. TTC TermSuite includes a graphical user interface tool with several embedded UIMA components that perform text and linguistic analysis up to monolingual term extraction and bilingual term alignment.

First, we present TTC TermSuite specifications that include the 4-step functional architecture in reverse order, the data model, and the input and output formats. Then, we detail the UIMA-based implementation, its components, the multilingualism management and the graphical interface for building tool chains easily. We conclude by the case study: the extraction of SWTs from a comparable corpora in two pairs of languages.

[1] http://uima.apache.org

Proceedings of the IJCNLP 2011 System Demonstrations, pages 9–12,
Chiang Mai, Thailand, November 9, 2011.

Functional Architecture	Required/Input data	Provided/Output data
Text Pre-Processing		text, language
Linguistic Analysis	text, language	word, part-of-speech lemma
word tokenization	text, language	word
part-of-speech tagging	language, word	part-of-speech
lemmatization	language, word, part-of-speech	lemma
Term Extraction	language, word, part-of-speech lemma	term
Term Alignment	language, term	binary relation over terms

Table 1: TTC TermSuite 4-step Functional Architecture & Data Model

2 Specifications

The TTC TermSuite specifications consist of the definition of functional computing units within an architecture, the data model shared between these units and the file formats of this data model. Table 1 summarizes the 4-step functional architecture, and the input and output data types for each functional step.

2.1 Functional Architecture

The functional architecture is divided into 4 steps: text pre-processing, linguistic analysis, monolingual term extraction, bilingual term alignment. A set of tools will be assigned to each step:

Text pre-processing web-crawlers, text categorizers, text extractors, data cleaning, language recognizers, etc. All tools that provide a clean textual content without any linguistic information.

Linguistic analysis word tokenizers, part-of-speech taggers, lemmatizers, morphological analyzers and syntactic parsers.

Term extraction single-word term (SWT), multi-word term (MWT) and morphological compound detection, term variant processing such as acronym detection;

Term alignment SWT and MWT alignment, cognate detection, machine translation on the fly for MWTs.

2.2 Data Model

The TTC TermSuite's 4-step architecture requires a data model that defines the data types required as input and output for each functional unit.

The output of the *text pre-processing* step should provide at a minimum the textual data of the document and the language it is written in. Textual data and language are required by the *linguistic analysis* step. According to the language, miscellaneous treatments are applied to the textual data that could be useful for the *term extraction* step such as part-of-speech and lemma taggers, morphological analysis. Part-of-speech and lemma are required for the *term extraction* step that performs both SWT and MWT extraction. The output of the *term extraction* step is a list of candidate terms that is required by the *term alignment* step. TTC TermSuite outputs one-to-many alignments: a source term associated to the set of its most probable target translations in the target language. It should be noticed that the first two steps deal with the document processing whereas the last two steps deal with the document collection processing.

2.3 Input and Output Formats

TTC TermSuite's input and output files are XML files which adopts standard formats. Document features are formatted according to the Dublin Core XML Schema. A Dublin Core input file with the location, the language, the format of the resource can be represented as follows:

```
<metadata>
  <language>english</language>
  <format>text/html</format>
  <title>Top Myths About Wind Energy</title>
  <source>
    http://www.bwea.com/energy/myths.html
  </source>
  <subject>
    wind energy, wind turbine, wind farm, wind power plant
  </subject>
</metadata>
```

Moreover, the terms that have led to *crawl* this document is also provided by the Dublin Core

subject element.

As for terminologies, they are formatted according to the TermBase eXchange XML Schema (TBX) [ISO 30042:2008] compliant with the TMF (Terminological Markup Framework) meta-model [ISO 16642:2001]. Such an output file with an alignment between English and Chinese for the term *wind energy* corresponds to the sample below:

```
<martif type="TBX">
  <text>
    <body>
      <termEntry id="term-entry-1">
        <langSet xml:lang="en">
          <tig>
            <term id="term-1">wind energy</term>
            <descrip type="alignment" target="term-16"/>
          </tig>
        </langSet>
      </termEntry>
      <termEntry id="term-entry-16">
        <langSet xml:lang="zh">
          <tig>
            <term id="term-16">风能</term>
            <descrip type="alignment" target="term-1"/>
          </tig>
        </langSet>
      </termEntry>
    </body>
  </text>
</martif>
```

Terms and term entries of the TermBase eXchange files provided by the TTC TermSuite can be enriched with other features such as the term constituent, their part-of-speech, their lemma, their different occurrences in the corpora, etc according to the linguistic analyzes that have been processed.

3 UIMA implementation

The UIMA-based implementation consists of components that can be easily aggregated together through a user-friendly graphical interface, are powered by the UIMA framework, and are designed to manage multilingualism.

3.1 Graphical Interface

With the TTC TermSuite, it is possible to design UIMA tool chains easily; users can create or open several tool chains. They can select their components merely by dragging them from the available ones and dropping them on the selected ones. Component metadata can be displayed by double clicking on an available component whereas component parameters can be set by double clicking on a selected one. There are TTC TermSuite panels for processing tool chains and viewing their results such as illustrated in the Figure 1.

3.2 UIMA Components

UIMA offers a common, standards-based software architecture facilitating reuse and integration, it solves essentially issues connected with lower-level interoperability of software components. UIMA main concepts are:

Collection Processing Engine (CPE) Tool chains are formalised by CPE within UIMA. They are defined by 1 Collection Reader and by 1 or more Analysis Engine.

Common Analysis Structure (CAS) UIMA adopts a common representation to represent any artifact being analyzed and to provide reading/writing access to the analysis results or annotations. CAS ensures CPE component interoperability thanks to a **Type System** that can be indexed in CAS.

Collection Readers are the only CPE components able to create CAS.

Analysis Engines are CPE components that produce structured information by indexing annotations in CAS.

Up to now more than 60 components are provided within the TTC TermSuite but 4 of them can be drawn out that corresponds to the 4 steps of the functional architecture. The first 2 steps are completed. Step 3 and 4 are still under development but are completed for SWTs.

1. **Text Preprocessing** is a Collection Reader creates CAS from Dublin Core metadata.
2. **Linguistic Analysis** is an Analysis Engine that detects words, their part-of-speech and their lemma.
3. **Term Extraction** is an Analysis Engine that adopts a homogeneous approach for both SWTs and MWTs. Terms are first extract thanks to morpho-syntactic patterns defined for each languages and rank according to statistical criteria (Daille, 2002).
4. **Term Alignment** is an Analysis Engine that aligns SWTs using a lexical context analysis (Morin et al., 2010)

UIMA components are provided through out a Google Code repository for managing Open-Source source code[2].

[2]http://code.google.com/p/ttc-project/

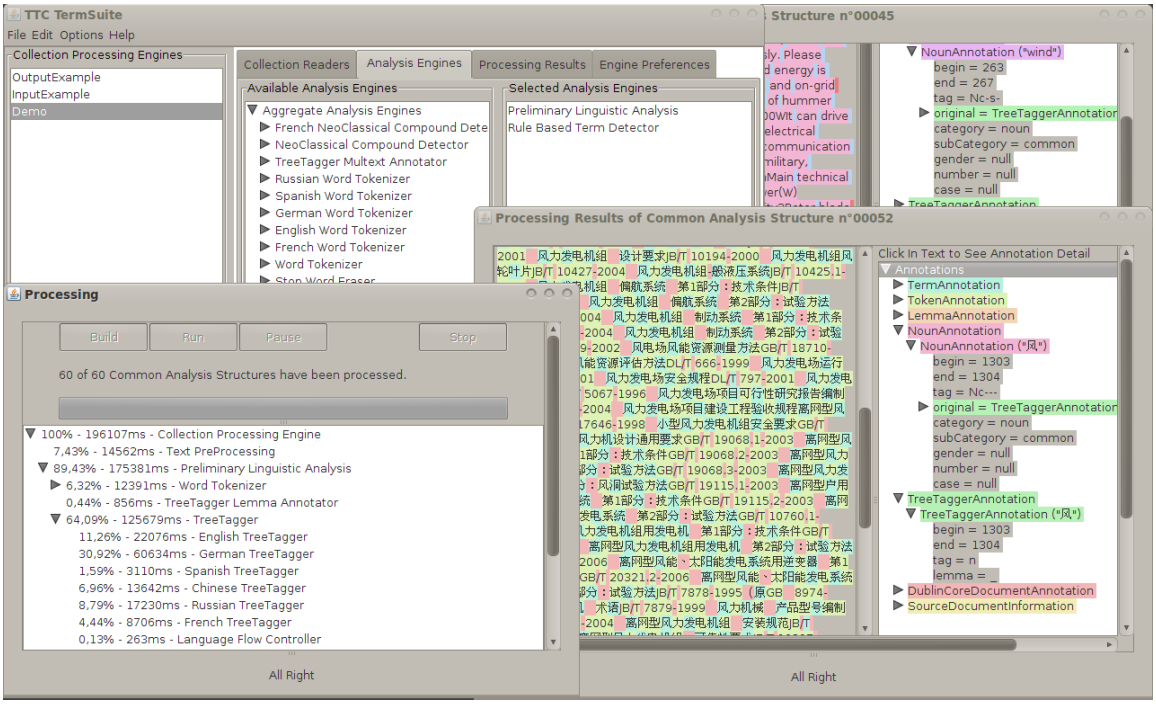

Figure 1: Graphical interface of TTC TermSuite

3.3 Multilingual Management

Multilingualism is delegated to CPE components level e.g. to Analysis Engines. As the language of the CAS is set by the Text PreProcessing Collection Reader and as each Analysis Engine specifies which languages they analyze, CAS can be dispatched to the corresponding AE.

4 Demonstration

The TTC TermSuite will be demonstrated using the following case study: it will extract SWTs from comparable corpora that deal with renewable energy for two pairs of languages: French-English and English-Chinese.

Acknowledgement

The research leading to these results has received funding from the European Communitys Seventh Framework Programme (*/*FP7/2007-2013*/*) under Grant Agreement no 248005.

References

[Bowker and Pearson2002] Lynne Bowker and Jennifer Pearson. 2002. *Working with Specialized Language: A Practical Guide to Using Corpora*. London/New York: Routledge.

[Daille2002] Béatrice Daille. 2002. Terminology mining. In Maria Teresa Pazienza, editor, *SCIE*, volume 2700 of *Lecture Notes in Computer Science*, pages 29–44. Springer.

[Ferrucci and Lally2004] David Ferrucci and Adam Lally. 2004. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10:327–348, September.

[Koehn2005] Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.

[Merkel and Foo2007] Magnus Merkel and Jody Foo. 2007. Terminology extraction and term ranking for standardizing term banks. In *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA-07)*, pages 349–354, Tartu.

[Morin et al.2010] Emmanuel Morin, Béatrice Daille, Koichi Takeuchi, and Kyo Kageura. 2010. Brains, not brawn: The use of ”smart” comparable corpora in bilingual terminology mining. *TSLP*, 7(1).

[Och and Ney2003] Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

[Rapp1995] Reinhard Rapp. 1995. Identify Word Translations in Non-Parallel Texts. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL'95)*, pages 320–322, Boston, MA, USA.