OntoRest: Text Mining Web Services in BioC Format

Hernani Marques*
hernani.marquesmadeira@uzh.ch

Fabio Rinaldi*
fabio.rinaldi@uzh.ch

Abstract

In this poster we present a set of biomedical text mining web services which can be used to provide remote access to the annotation results of an advanced text mining pipeline. The pipeline is part of a system which has been tested several times in community organized text mining competitions, often achieving top-ranked results.

1 Introduction

Web services play an increasingly important role in computational biology in general, and in biomedical text mining in particular. Recently emerged biomedical text mining services can provide results from tasks such as Named Entity Recognition (NER) or Relationship Recognition (RE) in an easily reusable format (Rak et al., 2012; Nunes et al., 2013). Such web services offer text mining capabilities to a wider public of potential users that do not necessarily have the skills or the time to install specialized software. Additionally, interoperability among different (web) services can be achieved.

With **OntoRest** (OntoGene RESTful) we provide a web service suite offering access to the annotation results of the **OntoGene** text mining pipeline, which is optimized for the biomedical text domain. Input material can be provided in the community-endorsed BioC XML format through a RESTful API, for it to be returned in this same format, enriched by the specific annotations requested.

In the task 3 of the BioCreative 2013 competition participants were requested to provide a web service capable of returning biomedical annotations of utility for the Comparative Toxicogenomics Database (Davis et al., 2011). The annotations had to be provided in a pre-defined community-specified XML format called BioC (Comeau et al., 2013). This file format is designed for the needs of biomedical text mining, but could in principle also be applied to other texts types. It provides both in-line and stand-off annotations: the former to allow the annotation of different text segments like passages or sentences; the later to allow for rich annotation of any piece of information found in a text. The stand-off annotations can be enriched through the provision of so-called `infon` ("information") elements. Besides of locational (`location` tag) and relational (`relation` tag), additional tags are available to specify which place is exactly annotated, but also which of the annotations may be interrelated.

In the course of the 2013 task organized by the BioCreative community our team created the initial version of **OntoRest** (Rinaldi et al., 2013) – the OntoGene RESTful annotation web service for biomedical text mining. In recent months the services have been overhauled and extended, as discussed in this paper. In section 2 we present in general terms the design and implementation of OntoRest. In section 3 we describe its functions and features.

2 System design and implementation

Primarily, the idea of OntoRest is to adhere to a simple input–output client–server model through which biomedical text mining can be carried out using the underlying OntoGene biomedical text mining pipeline on a collection basis, which means that one or more documents can be submitted for annotation, using the BioC XML format.

*University of Zurich, Institute of Computational Linguistics, Binzmühlestr. 14, 8050 Zürich, Switzerland

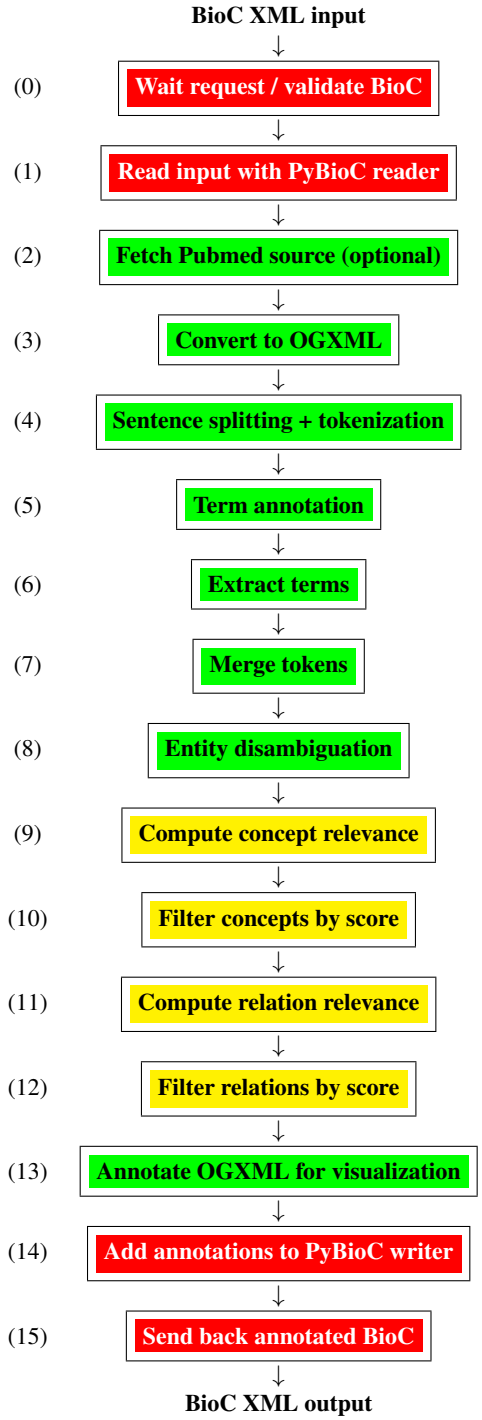

Figure 1: **OntoRest and OntoGene pipeline**: green boxes represent the original OntoGene pipeline, yellow boxes represent additions that optimize the ranking of concepts and relations, red boxes represent OntoRest proper.

BioC is a recently proposed annotation format which aims at allowing a simplified data exchange among text mining systems or their components. BioC comes with a light specification of an XML format (Comeau et al., 2013), plus a set of tools which implement the standard in various programming languages (Liu et al., 2014). Our group provided a native Python implementation called PyBioC (Marques and Rinaldi, 2013).

OntoGene is a text mining pipeline which makes heavy use of the UNIX pipeline philosophy, i.e. conversion of file formats, tokenization etc. are done in a sequential way following a clear order: each program involved has a clearly defined and limited task on its own. Intermediate output files are typically either XML files preserving the original document structure or TSV files (tab separated values), which contain some of the mined results. For each single step performed by the OntoGene pipeline, log files are generated, such that – in contrast to heavily integrated systems – problems can be easily localized. OntoRest uses some of the intermediate files produced by the OntoGene pipeline to deliver a response in BioC XML format.
The OntoRest service performs the following tasks:

- listen for RESTful POST requests on a designed port
- when a request is received, check it for validity, and reject it with an error message if not valid
- if the request is valid, convert the input BioC XML into the input format of the OntoGene pipeline
- invoke the OntoGene pipeline with the converted file as input
- collect the results of the pipeline, and convert it back into BioC
- send the final BioC document, enriched with the results of the text mining pipeline, to the requester

The OntoRest system makes use of the Representational State Transfer (REST) architecture (Richardson and Ruby, 2007) built upon the World Wide Web most important protocol – the Hyper

Text Transfer Protocol (HTTP). This means that the system interface consists of a web server acting upon HTTP requests of client systems. However, those clients can be web applications (being web services on their own) which use the output of OntoRest to convert the annotations it delivers in another mark-up language (like HTML) or in any other format for further processing.

OntoRest is implemented entirely in Python, tested to work under version 2.7.6 onwards. It makes use of the following non-standard libraries to implement its functionality:

- The Twisted library for the lower-level TCP/IP interaction (`https://twistedmatrix.com/trac/`).
- The Bottle library for handling RESTful requests, and the implementation of the surface parts of the HTTP server (`http://bottlepy.org/`).
- The PyBioC library to provide interoperability in terms of file formats (`https://github.com/2mh/PyBioC`). PyBioC is a native Python implementation of the BioC architecture created by the OntoGene team (Marques and Rinaldi, 2013) during the 2013 BioCreative challenge. This library features readers and writers for handling, manipulating and creating files in BioC XML format.

3 System features and functioning

To use OntoRest, a user has to submit a document in valid BioC XML format, as specified by the corresponding Document Type Definition (DTD) file. The document is sent in a RESTful way: that is by using the HTTP POST method towards a certain URL (with a port) e.g. `http://locahost:8081/ctd`. Currently the services are available from the base URL:
`https://pub.cl.uzh.ch/projects/BioC/`

The web service provides six request handlers (see table 1), which can either be used for basic system tests (using the `/iotest` request handler), or to obtain annotations on any user-provided textual material (as long as it is in the required BioC format). The system provides annotations for

Request handler	**Output delivered**
`/iotest`	Input XML without modifications
`/chem`	Input XML annotated for chemicals
`/disease`	Input XML annotated for diseases
`/gene`	Input XML annotated for genes
`/ctd`	Input XML with all annotations (gene, chemicals, diseases)
`/ixn`	Input XML with interaction terms

Table 1: Request handlers supported by OntoRest.

chemicals, diseases or genes, using as entity identifiers those use by the CTD database. Using the `/ctd` request handler it is possible to easily get annotations for chemicals, diseases and genes in one run (this is one of the recent additions which was not part of the BioCreative IV challenge). In order to use a specific service, the user has to combine the basic URL with the specific request handler, e.g. use `https://pub.cl.uzh.ch/projects/BioC/disease` to obtain the mentions of diseases in the input document. The list of services might be expanding as we integrate them with the existing OntoGene pipeline. An up-to-date list can be found at
`http://www.ontogene.org/webservices/`

For the BioCreative task only concept identifiers (according to specific biomedical databases) were supposed to be made available. However, following the challenge, further `infon` elements were added for the NER sub-task of detecting chemicals, diseases and genes, which currently are:

- By default OntoRest does not just provide the concept name, but also the surface form found in the text.
- Providing a suffix like the implemented `+offset` attached to either `/ctd`, `/chem`, `/disease` or `/gene` when calling the web service designates an optional argument. In case of the `+offset` option, a location element is added to the annotation tags making clear where an annotation starts and how long it is.
- Using the option `+rel` (for "relations"), the top 10 annotations are returned showing which

HTTP error code	Possible problems
400 Bad Request	Invalid data (not BioC XML) was submitted.
404 Not Found	An invalid URL was used (including nonexistent options).
405 Method Not Allowed	A wrong HTTP method was used (typically `GET` instead of `POST`).

Table 2: Request handlers supported by OntoRest.

concepts appear together along with its score of relatedness, by which they are sorted. This currently can be applied only to the `/ctd` request handler involving chemicals, diseases and genes.

For example, a request with the URL
`https://pub.cl.uzh.ch/projects/BioC/gene+offset`
will deliver all genes found in the input document with their positions in the original text. A request with the handler `/ctd+rel` delivers all entities detected in the input document and the 10 top-ranked interactions. A forthcoming extension will allow the user to select the number of interactions that they want to be delivered. Another planned addition will allow the user to select interactions among specific entity types (e.g. gene-disease only).

OntoRest strictly accepts only BioC XML text as input, which must be transferred via the HTTP POST method, and generates as output an enriched version of the input document, conforming to the BioC specification, and containing annotations provided by the OntoGene pipeline.

In cases where no valid input material or no allowed HTTP methods are used, HTTP errors are triggered to signalize wrong service usage. The possible errors and their common underlying problems are illustrated in table 2. The current version of the web services is publicly accessible as described in `http://www.ontogene.org/webservices/` Figure 2 shows an example of output document.

4 Conclusions

We presented a set of web services (OntoRest) which provide advanced text mining capabilities to remote users. The services rely on the recently proposed and rapidly spreading BioC standard for data interchange in the biomedical text mining community. OntoRest has been originally designed as part of the BioCreative 2013 challenge, and has been recently extended to provide more advanced services. We continue its development and intend to provide soon additional functionalities.

References

Donald C. Comeau, Rezarta Islamaj Doan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifang Peng, Fabio Rinaldi, Manabu Torii, Alfonso Valencia, Karin Verspoor, Thomas C. Wiegers, Cathy H. Wu, and W. John Wilbur. 2013. BIoC: a minimalist approach to interoperability for biomedical text processing. *The Journal of Biological Databases and Curation*, bat064. published online.

AP Davis, BL King, S Mockus, CG Murphy, C Saraceni-Richards, M Rosenstein, T Wiegers, and CJ Mattingly. 2011. The comparative toxicogenomics database: update 2011. *Nucleic Acids Res.*, 39(Database issue):D1067–72, Jan.

Wanli Liu, Rezarta Islamaj Doan, Dongseop Kwon, Hernani Marques, Fabio Rinaldi, W. John Wilbur, and Donald C. Comeau. 2014. BioC implementations in Go, Perl, Python and Ruby. *Database: The Journal of Biological Databases and Curation*, bau059.

Hernani Marques and Fabio Rinaldi. 2013. PyBioC: a python implementation of the bioc core. In *Proceedings of the Fourth BioCreative Challenge Evaluation Workshop*, volume 1, pages 2–4.

Tiago Nunes, David Campos, Srgio Matos, and Jos Lus Oliveira. 2013. BeCAS: biomedical concept recognition services and visualization. *Bioinformatics*, 29(15):1915–1916.

Rafal Rak, Andrew Rowley, William Black, and Sophia Ananiadou. 2012. Argo: an integrative, interactive, text mining-based workbench supporting curation. *Database*, 2012.

Leonard Richardson and Sam Ruby. 2007. *RESTful Web Services*. O'Reilly, Sebastopol, California. ISBN 978-0-596-52926-0.

Fabio Rinaldi, Simon Clematide, Tilia Renate Ellendorff, and Hernani Marques. 2013. OntoGene: CTD entity and action term recognition. In *Proceedings of the Fourth BioCreative Challenge Evaluation Workshop*, volume 1, pages 90–94.

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE collection SYSTEM 'BioC.dtd'>
<collection>
  <source>PUBMED</source>
  <date>20100202</date>
  <key>ctdBCIVLearningDataSet.key</key>
  <document>
    <id>20130422</id>
    <passage>
      <infon key="type">title</infon>
      <offset>0</offset>
      <text>Chemotherapy resistance as a predictor of progression-free
survival in ovarian cancer patients treated with neoadjuvant
chemotherapy and surgical cytoreduction followed by intraperitoneal
chemotherapy: a Southwest Oncology Group Study.</text>
    </passage>
    <passage>
      <infon key="type">abstract</infon>
      <offset>104</offset>
      <text>In vitro testing of the activity of chemotherapeutic agents
has been suggested as 1 method to optimally select drugs for patients
with ovarian cancer. There are limited prospectively obtained data
examining the clinical utility of this approach. We sought to obtain a
preliminary assessment of this strategy in a trial that examined the
administration of neoadjuvant chemotherapy followed by surgical
cytoreduction and intraperitoneal chemotherapy in women with advanced
ovarian cancer. Women with stage III/IV epithelial ovarian carcinoma
that presented with large-volume disease were treated with neoadjuvant
intravenous paclitaxel and carboplatin for three 21-day cycles followed
by cytoreductive surgery. If optimally debulked, patients received
intravenous paclitaxel, intraperitoneal carboplatin and intraperitoneal
paclitaxel for six 28-day cycles. Tumor cloning assay results (Oncotech)
were correlated with progression-free survival. Sixty-two patients (58
eligible) were registered from March 2001 to February 2006. Thirty-six
eligible patients had interval debulking and 26 received
postcytoreduction chemotherapy. Twenty-two patients had tumor cloning
assay results available. The clinical features of this population were
similar to those of the larger group of women who entered this study.
There was no difference in progression-free survival between patients
whose cancers were defined as 'resistant' or 'nonresistant' to either
platinum or paclitaxel. While the small patient numbers in this trial do
not permit definitive conclusions, these data fail to provide support
for the argument that prospectively obtained in vitro data regarding
platinum or paclitaxel resistance will be highly predictive of clinical
outcome in advanced ovarian cancer.</text>
      <annotation>
        <infon key="type">chem</infon>
        <infon key="id">MESH_D016190</infon>
        <text>Paclitaxel</text>
      </annotation>
      <annotation>
        <infon key="type">chem</infon>
        <infon key="id">MESH_D017239</infon>
        <text>Carboplatin</text>
      </annotation>
      <annotation>
        <infon key="type">disease</infon>
        <infon key="id">MESH_D010051</infon>
        <text>Ovarian Neoplasms</text>
      </annotation>
    </passage>
  </document>
</collection>
```

Figure 2: **Example of output of OG text mining service in BioC format.** The output of the system is generated in the BioC specification format. This oputput was generated by querying for chemicals and diseases on PubMed abstract 20100202. Colors added for clarity. Offsets of annotated terms can be obtained with a separate query.