Construction of a biodiversity knowledge repository using a text mining-based framework

Riza Batista-Navarro, **Chrysoula Zerva** and **Sophia Ananiadou**

School of Computer Science
University of Manchester
Manchester, United Kingdom
{riza.batista,chrysoula.zerva,sophia.ananiadou}@manchester.ac.uk

Abstract

In our aim to make the information encapsulated by biodiversity literature more accessible and searchable, we have developed a text mining-based framework for automatically transforming text into a structured knowledge repository. A text mining workflow employing information extraction techniques, i.e., named entity recognition and relation extraction, was implemented in the Argo platform and was subsequently applied on biodiversity literature to extract structured information. The resulting annotations were stored in a repository following the emerging Open Annotation standard, thus promoting interoperability with external applications. Accessible as a SPARQL endpoint, the repository supports knowledge discovery over a huge amount of biodiversity literature by retrieving annotations matching user-specified queries.

1 Introduction

Big data—huge data collections—are proliferating in many disciplines at a rate that is much faster than what our analytical abilities can handle. One particular discipline that has amassed big data is biological diversity, more popularly known as biodiversity: the study of variability amongst all life forms. On the one hand, researchers in this domain collect primary data pertaining to the occurrence or distribution of species, and store this information in a structured format (e.g., spreadsheets, database tables). On the other hand, findings or observations resulting from their analysis of primary data are usually reported in literature (e.g., monographs, books, journal articles or reports), often referred to as secondary data. Written in natural language, secondary data lacks the structure that primary data comes with, rendering the knowledge it contains obscured and inaccessible. In order to make information from secondary data available in a structured and thus searchable form, we have developed a repository containing information automatically extracted from biodiversity literature by a customisable text mining workflow. To maximise its interoperability with external tools or services, we have made the knowledge repository available as a Resource Description Framework (RDF) triple store that conforms with the Open Annotation standard[1]. We then demonstrate how the repository, accessible as a SPARQL endpoint, facilitates query-based search, thus making the information contained in biodiversity literature discoverable.

A handful of other tools for storing biodiversity information in RDF format exist. Most of them, however, do not have the capability to automatically understand text written in natural language. Tools such as RDF123 (Han et al., 2008) and BiSciCol Triplifier (Stucky et al., 2014), for example, accept only data that is already in the form of structured tables. The browser extension Spotter (Parr et al., 2007) generates RDF-formatted annotations over blog posts, not by automatically extracting information from the textual content but rather by requiring its users to manually enter structured descriptive metadata. Most similar to our work is a system for automatically extracting RDF triples pertaining to species' morphological characteristics, from the literature on Flora of North America (Cui et al., 2010). Their semantic annotation application provided the user with an opportunity to revise automatically generated annotations, an option that can also be enabled in our approach. We note though that our work

[1] http://www.openannotation.org

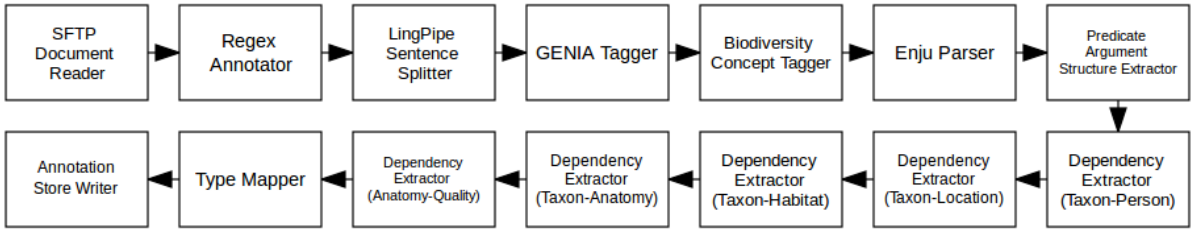

Figure 1: Text mining workflow

is uniquely underpinned by a highly customisable and extensible workflow. In this way, when domain experts call for other types of information to be captured, our framework will require only minimal development time and effort to fulfill the task.

2 Methodology

In this section, we present in detail our framework for constructing the knowledge repository. We begin by briefly describing the corpus of biodiversity documents that was utilised, and then outline the various steps in the text mining workflow. We finally proceed to explaining how the Open Annotation specification was adopted in order to store the information extracted from our corpus.

2.1 Document selection

The Biodiversity Heritage Library (BHL)[2] is a database of biodiversity literature maintained by a consortium of natural history and botanical libraries all over the world. A product of the various partners' digitisation efforts, BHL currently contains almost 110,000 titles, equivalent to almost 50 million pages of text resulting from the application of optical character recognition (OCR) tools on scanned images of legacy materials. For this work, we decided to narrow down the scope of the knowledge repository to the requirements of our ongoing project whose aim is to comprehensively collect both primary and secondary information on biodiversity in the Philippines.

To this end, we retrieved only the subset of English BHL pages which are relevant to the Philippines, i.e., the union of (1) the set of pages which mention either "Philippines" or "Philippine" within their content, and (2) the set of pages contained by books or volumes whose titles mention "Philippines" or "Philippine". This resulted in a corpus of a total of 155,635 pages (around 12GB in size).

[2]http://www.biodiversitylibrary.org

2.2 Development of text mining workflow

One of the primary interests of our collaborators in the project is the discovery of fundamental species-centric knowledge, particularly information on species' geographic locations, habitat, anatomical parts as well as authorities (i.e., persons who described them). Guided by user requirements, we cast this work as an information extraction task requiring: (1) named entity recognition (NER) for taxa, locations, habitat, anatomical parts and persons; and (2) binary relation extraction focussing on the following types of associations: taxon-location, taxon-habitat, taxon-anatomical part and taxon-person.

To carry out these tasks on our corpus, we integrated various natural language processing (NLP) tools into one workflow using the Argo platform. Argo[3] is a web-based, graphical workbench that facilitates the construction and execution of bespoke modular text mining workflows. Underpinning it is a library of diverse elementary NLP components, each of which performs a specific task. Argo's graphical block diagramming interface for workflow construction provides access to the component library, representing them as configurable blocks that can be interconnected to define processing sequence.

The workflow that we developed, depicted in Figure 1, combines several components for preprocessing, synactic and semantic analyses. It begins with an SFTP Document Reader which loads the plain-text corpus from a remote server. This is followed by a Regex Annotator which attempts to detect paragraph boundaries based on the occurrence of newline characters. The paragraphs are then segmented by the LingPipe Sentence Splitter[4] into sentences, each of which is decomposed into tokens by the GENIA Tagger (Tsuruoka et al., 2005) which also performs part-of-speech tag-

[3]http://argo.nactem.ac.uk
[4]http://alias-i.com/lingpipe

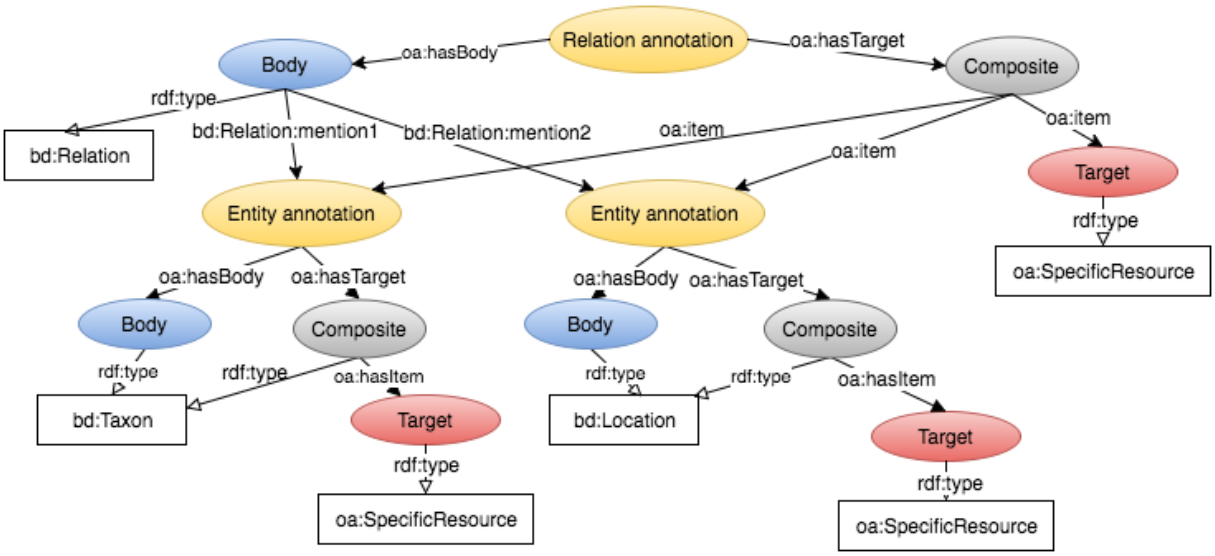

Figure 2: Our Open Annotation representation of related entities

ging, lemmatisation and chunking. The next component, the Biodiversity Concept Tagger, is a machine learning-based NER[5] that applies a conditional random fields (CRF) model (Lafferty et al., 2001) to assign labels to token sequences. The labels in this case correspond to the following categories: taxon, location, habitat, anatomical part, quality and person.

The succeeding components in the workflow contribute towards the relation extraction task. Enju Parser performs deep syntactic parsing and extracts syntactic dependencies amongst sentence tokens. Its outputs are used by the next component, the Predicate Argument Structure Extractor, to compute semantic dependencies in the form of predicate-argument structures. The five instances of the Dependency Extractor component then makes use of the predicate-argument structures to detect relationships between names categorised under the specified entity types. The first instance, for example, detects only relationships between taxon and person names, while the last one captures related anatomical parts and qualities. The Type Mapper ensures that all of the named entities and relations extracted conform with the same annotation schema before they are all saved in Open Annotation format by the last component, the Annotation Store Writer. We briefly describe next how our extracted annotations are encoded according to this format.

[5]http://nersuite.nlplab.org

2.3 Adopting the Open Annotation model

The Open Annotation (OA) Core Data Model is an emerging W3C-recommended standard for encoding associations between any annotation and resource (i.e., what is being annotated). Built upon the Resource Description Framework (RDF), the OA model represents an annotation as having a *body* and a *target*, with the former somehow describing the latter, e.g., by assigning a label or identifier. Following this fundamental idea and other relevant recommendations given in the specification[6], we represented the named entity and relation annotations extracted by our text mining workflow in OA format, as depicted in Figure 2. For brevity, prefixes were used in this figure instead of full namespaces, e.g., `oa` for `http://www.w3.org/ns/oa#`.

Once the RDF triples had been generated, they were automatically loaded onto a new Apache Jena TDB[7] store, which was then exposed as a SPARQL endpoint by Fuseki[8].

3 Example use case

We present an example of how our repository, now in the form of a SPARQL-enabled triple store, can facilitate knowledge discovery. A user might be interested, for example, in learning which specific geographic locations have been described in the literature as having associations with certain

[6]http://www.openannotation.org/spec/core
[7]https://jena.apache.org/documentation/tdb
[8]https://jena.apache.org/documentation/fuseki2

species, e.g., the bird family of hornbills. Shown in Listing 1 is a query in SPARQL, the query language for RDF, that retrieves a list of all such locations, as well as the number of times that the relationship was mentioned in the source document.

Listing 1: An example SPARQL query that will retrieve locations related to hornbills.

```
PREFIX rdfs: <http://www.w3.org/2000/01/
    rdf-schema#>
PREFIX oa: <http://www.w3.org/ns/oa#>
PREFIX rdf: <http://www.w3.org
    /1999/02/22-rdf-syntax-ns#>
PREFIX bd: <http://nactem.ac.uk/schema/
    uima/typesystem/
    MiningBiodiversityTypeSystem#uk.ac.
    nactem.uima.biodiv.>
SELECT ?tx ?lc (COUNT(?lc) as ?cnt) ?src
WHERE {
 ?annotation oa:hasBody ?body .
 GRAPH ?body {
  ?a rdf:type bd:Relation .
  ?a bd:Relation:mention1 ?mention1 .
  ?a bd:Relation:mention2 ?mention2 . }
 ?mention1 oa:hasTarget ?target1 .
 GRAPH ?comp1 {
  ?target1 rdf:type bd:Taxon . }
 ?target1 oa:hasSelector ?selector1 .
 ?selector1 oa:default ?d1 .
 ?d1 oa:exact ?tx .
 FILTER(regex(?tx, "Hornbill", "i")) .
 ?mention2 oa:hasTarget ?target2 .
 GRAPH ?comp2 {
  ?target2 rdf:type bd:Location . }
 ?target2 oa:hasSelector ?selector2 .
 ?selector2 oa:default ?d2 .
 ?d2 oa:exact ?lc .
 ?target2 oa:hasSource ?src . }
GROUP BY ?tx ?lc ?src
ORDER BY DESC (?cnt)
```

4 Conclusion

In this paper, we presented a framework for building a knowledge repository that: (1) applies a customisable text mining workflow to extract information in the form of named entities and relationships between them; (2) stores the automatically extracted knowledge as RDF triples compliant with the Open Annotation specification; and (3) facilitates the discovery of otherwise obscured knowledge by enabling query-based retrieval of annotations from a SPARQL endpoint. We note that the triple store can be exposed via other application programming interfaces, i.e., web services that abstract away from SPARQL to make querying straightforward for non-technical users.

We envision that our knowledge repository will facilitate the enhancement of search applications, e.g., information retrieval systems. It has been made accessible as a SPARQL endpoint[9] that accepts POST requests. The body of the request should be set to a valid SPARQL query while the headers should be configured to hold the following name-value pairs: (1) Accept: text/csv and (2) Content-Type: application/sparql-query.

Acknowledgments

We would like to thank Prof. Marilou Nicolas for her valuable inputs. This work is funded by the British Council [172722806 (COPIOUS)], and is partially supported by the Engineering and Physical Sciences Research Council [EP/1038099/1 (CDT)].

References

Hong Cui, Kenneth Jiang, and Partha Pratim Sanyal. 2010. From Text to RDF Triple Store: An Application for Biodiversity Literature. In *Proceedings of the Association for Information Science and Technology (ASIST 2010)*.

Lushan Han, Tim Finin, Cynthia Parr, Joel Sachs, and Anupam Joshi. 2008. RDF123: From Spreadsheets to RDF. In Amit Sheth et al., editors, *Proceedings of the 7th International Semantic Web Conference (ISWC 2008)*, pages 451–466. Springer Berlin Heidelberg, Berlin, Heidelberg.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning (2001)*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Cynthia Parr, Joel Sachs, Lushan Han, and Taowei Wang. 2007. RDF123 and Spotter: Tools for generating OWL and RDF for biodiversity data in spreadsheets and unstructured text. In *Proceedings of Biodiversity Information Standards Annual Conference (TDWG 2007)*.

Brian J. Stucky, John Deck, Tom Conlin, Lukasz Ziemba, Nico Cellinese, and Robert Guralnick. 2014. The BiSciCol Triplifier: bringing biodiversity data to the Semantic Web. *BMC Bioinformatics*, 15(1):1–9.

Y. Tsuruoka, Y. Tateisi, J.-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. 2005. Developing a Robust Part-of-Speech Tagger for Biomedical Text. In *Advances in Informatics - 10th Panhellenic Conference on Informatics*, volume 3746 of *Lecture Notes in Computer Science*, pages 382–392. Springer-Verlag, Volos, Greece, November.

[9] http://nactem.ac.uk/copious-demo/annotations/sparql