Text mining workflows for indexing archives with automatically extracted semantic metadata

Riza Batista-Navarro[1], **Axel Soto**[1],
William Ulate[2] and Sophia Ananiadou[1]
[1]University of Manchester
[2]Missouri Botanical Garden

20th International Conference on Theory and Practice of Digital Libraries (TPDL 2016)

Outline (1)

- Introduction
  – challenges in information discovery/search
  – semantic metadata generation as a named entity recognition (NER) task
- The Argo text mining workbench
  – system overview and features
  – workflow construction, configuration and execution
  – visual inspection of generated annotations

Outline (2)

- Constructing NER workflows for generating semantic metadata
  - medical history archives
  - biodiversity legacy literature
- Exploring search indexes containing semantic metadata
  - introduction
  - overview of Elasticsearch
  - query examples

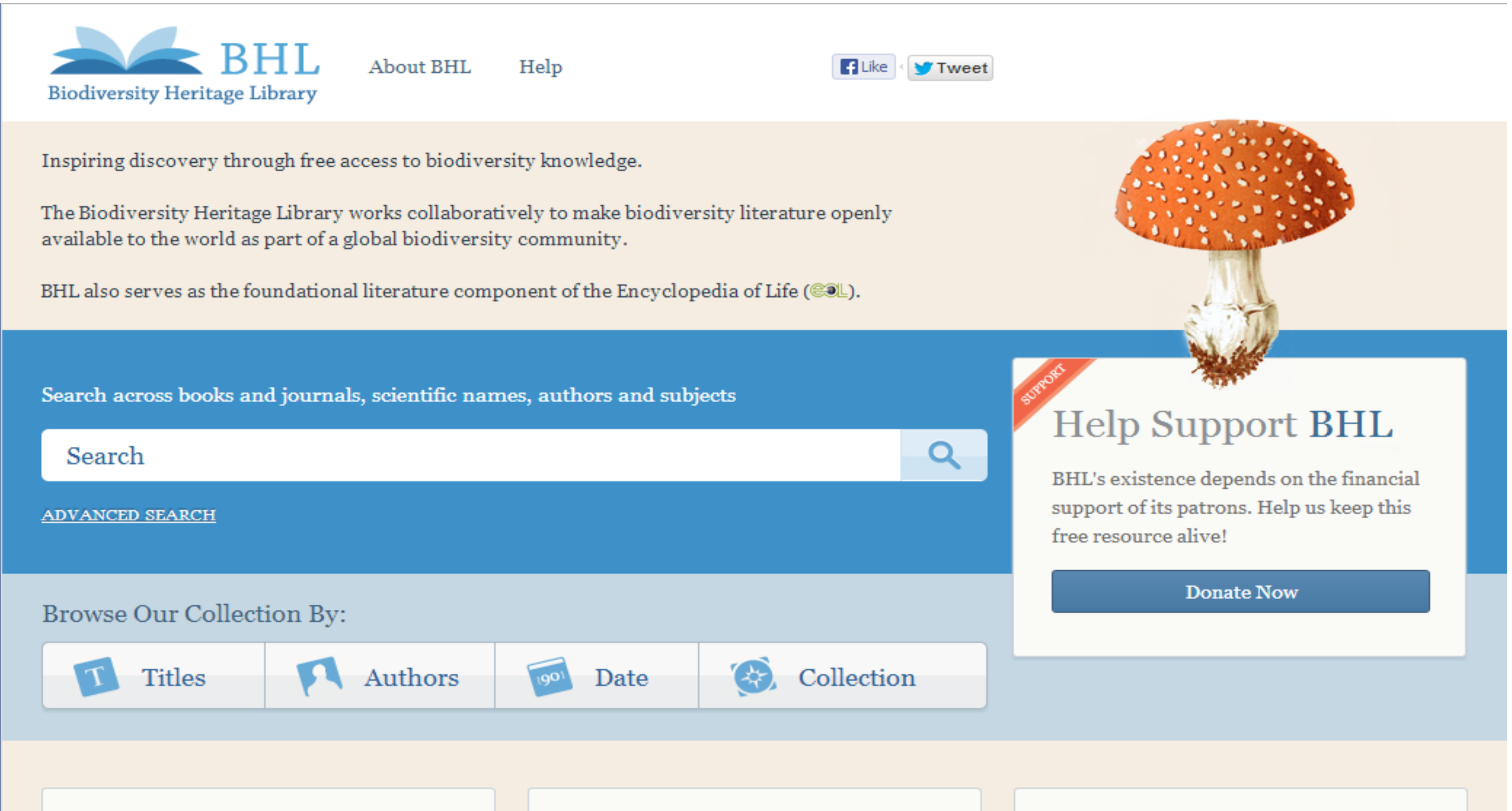
Outline (3)

- Example applications
  - Disambiguation in the History of Medicine search system
  - Biodiversity Heritage Library query expansion
- Conclusions

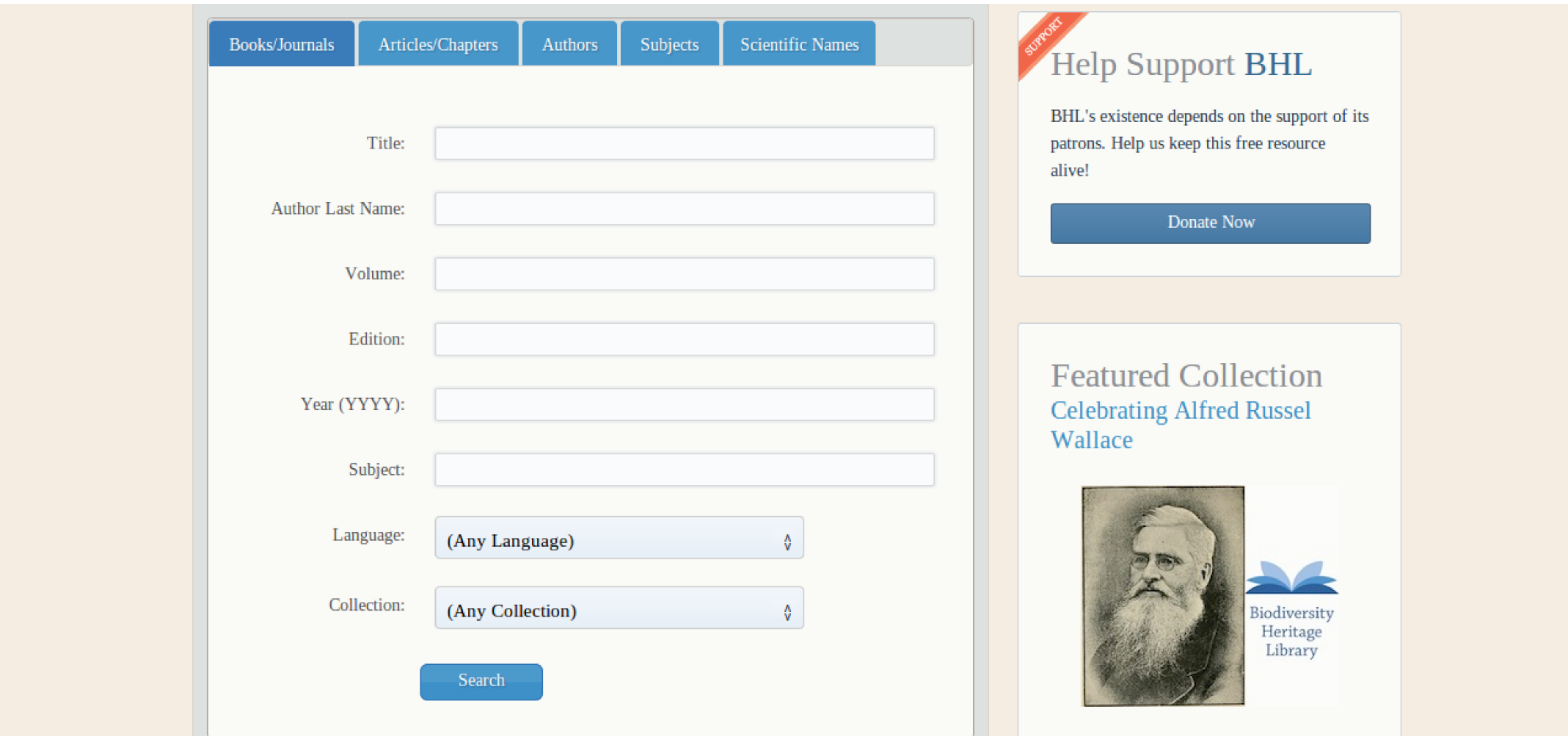
Biodiversity Heritage Library

- http://www.biodiversitylibrary.org/
- a consortium of botanical and natural history libraries
- stores digitised legacy literature on biodiversity
- currently holds 180,000 volumes = 50+ million pages (PDFs and OCR-generated text)
- open-access

BHL's keyword-based search and browsing

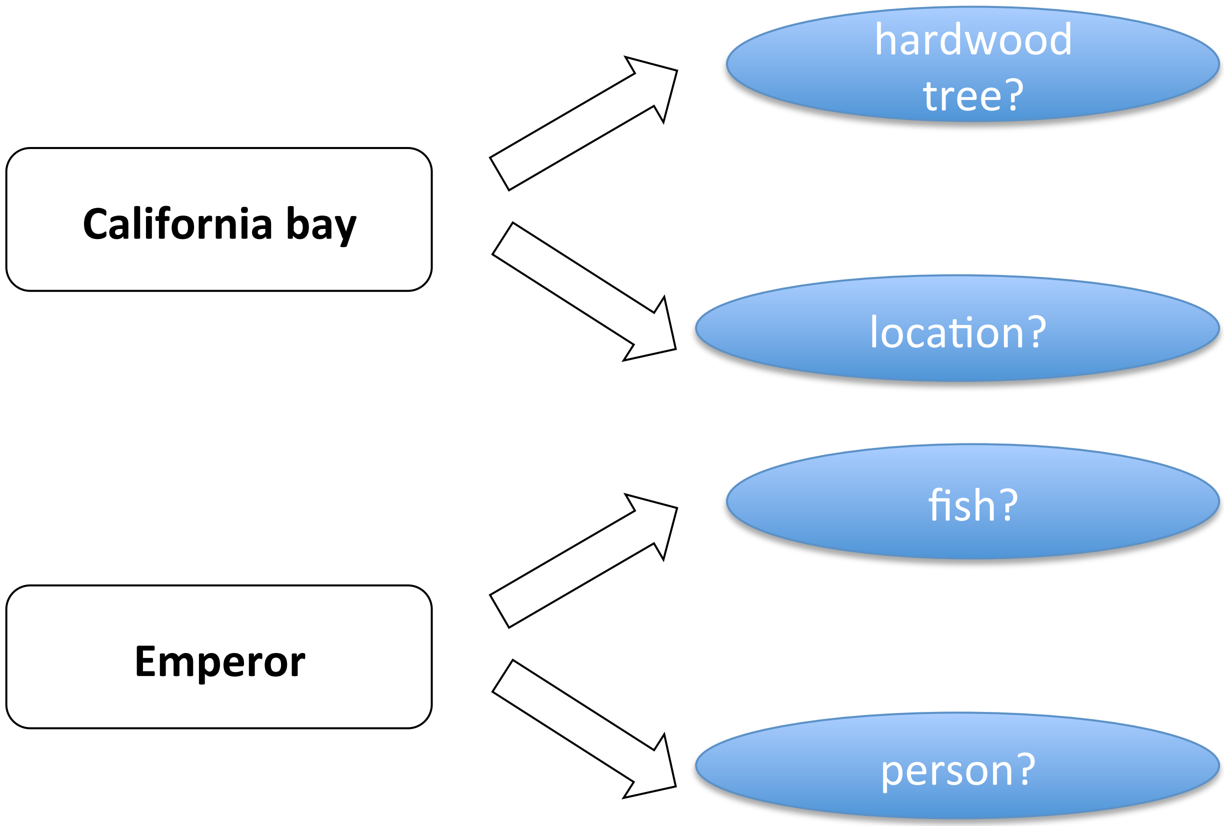


BHL's advanced search functionality (also keyword-based)



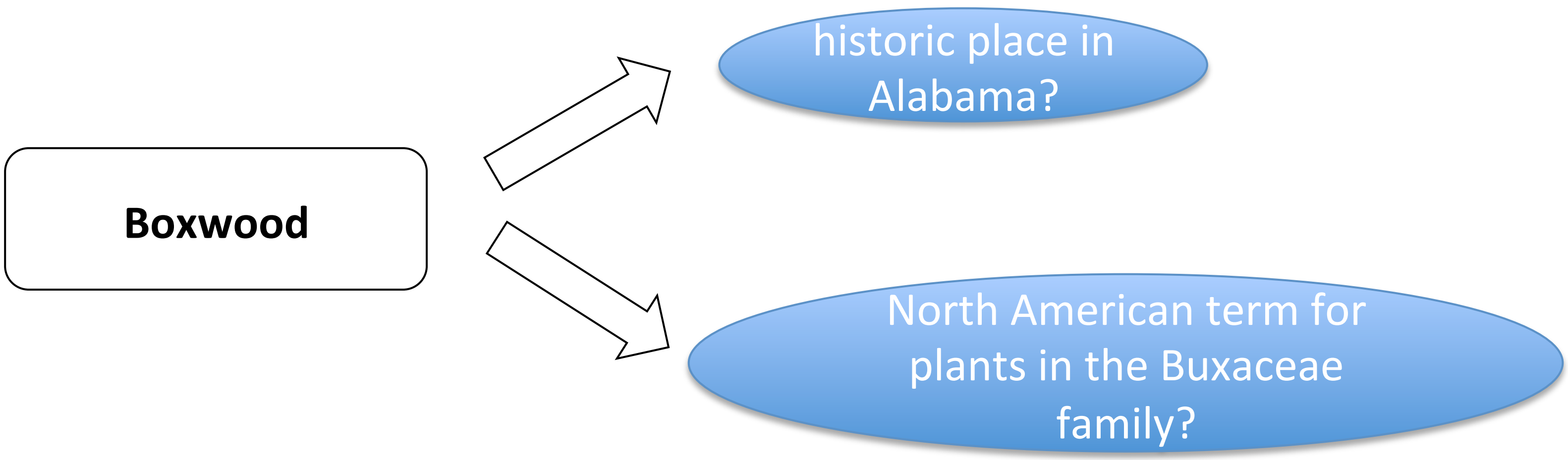
What's wrong with keyword-based search?

(1) Inability to disambiguate



What's wrong with keyword-based search?

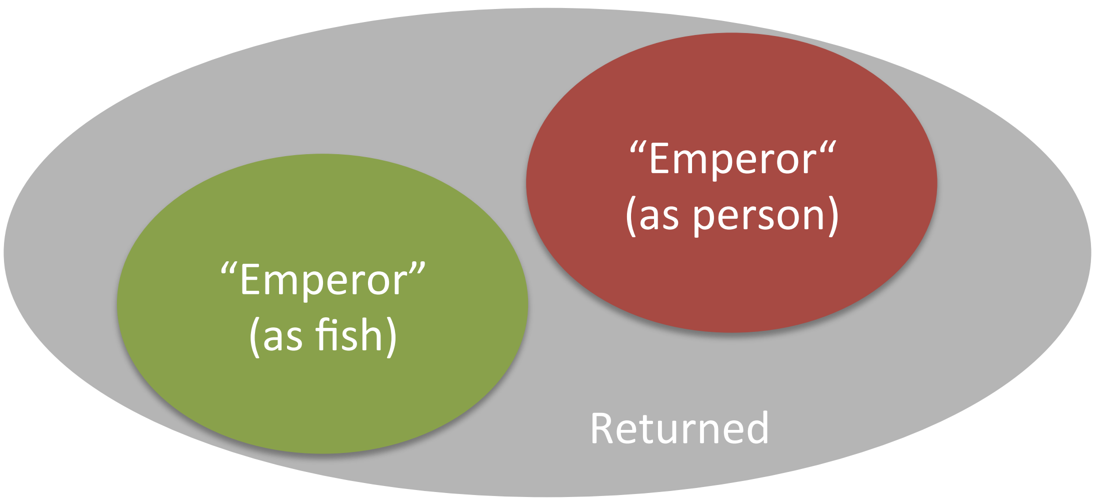
(1) Inability to disambiguate



What's wrong with keyword-based search?

(1) Inability to disambiguate

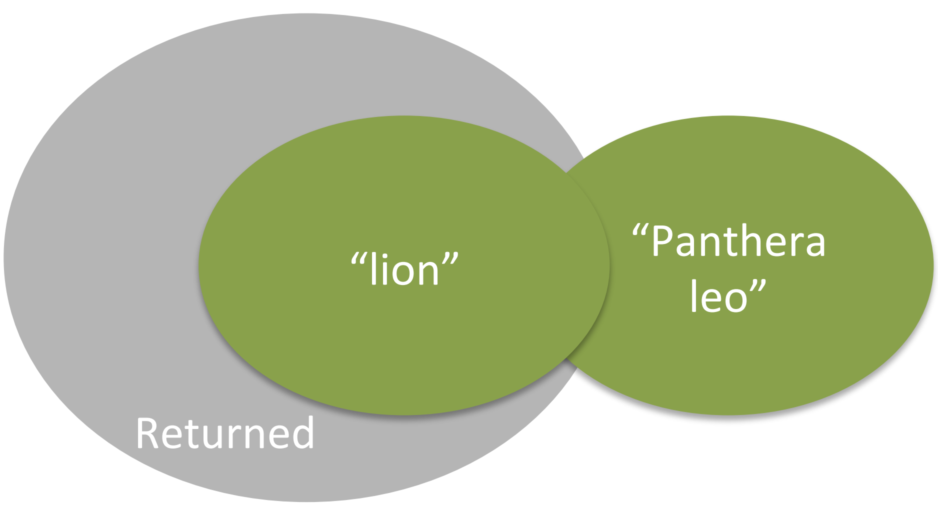
- Implications:
  - less precise search results
  - even documents irrelevant to what you have in mind are returned



What's wrong with keyword-based search?

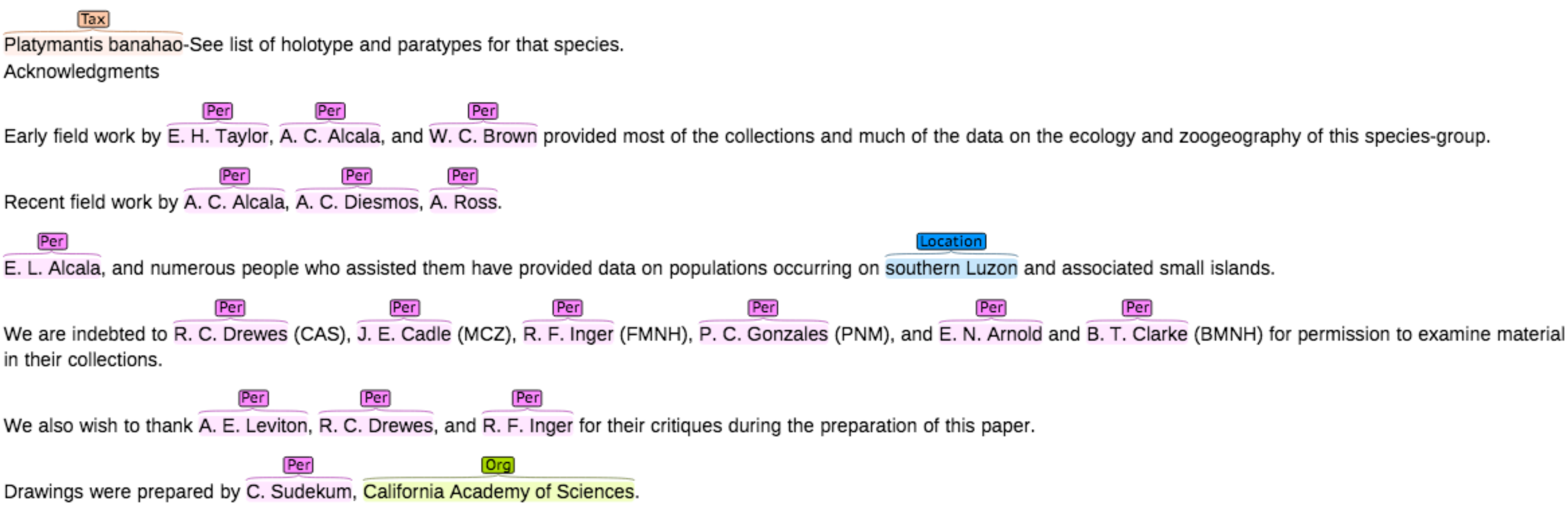
(2) Inability to account for variants

- Implications:
  - limited coverage
  - information overlook



Solution: Semantic metadata generation using named entity recognition (NER)

- task of automatically demarcating mentions
  - detecting their boundaries (e.g., character offsets)
  - placing them into predefined categories



Named entity recognition (NER)

- cast as a sequence labelling task
  - sequence = tokens in a sentence
- approaches
  - dictionary-based
  - rule-based
  - machine learning (ML)-based
  - hybrid

Dictionary-based NER

- **Sample entries in a gazetteer:**

Ho Chi Minh	PROVINCE
Ho Chi Minh City	CITY
…	
Johannesburg	CITY
Johannesburg	PROVINCE
…	
Mexico	PROVINCE
Mexico Beach	CITY
Mexico City	CITY
Mexico Crossing	CITY
…	
Riyadh	CITY
…	
Tehran	CITY
Tehran	PROVINCE

- **Sample text**

The
final
five
include
Mexico
City
,
Riyadh
,
Johannesburg
,
Ho
Chi
Minh
City

Dictionary-based NER

- **Sample entries in a gazetteer:**

Ho Chi Minh	PROVINCE
Ho Chi Minh City	CITY
…	
Johannesburg	CITY
Johannesburg	PROVINCE
…	
Mexico	PROVINCE
Mexico Beach	CITY
Mexico City	CITY
Mexico Crossing	CITY
…	
Riyadh	CITY
…	
Tehran	CITY
Tehran	PROVINCE

- **Sample text matched (in BIO)**

The	O	O
final	O	O
five	O	O
include	O	O
Mexico	B-CITY	B-PROVINCE
City	I-CITY	O
,	O	O
Riyadh	B-CITY	O
,	O	O
Johannesburg	B-CITY	B-PROVINCE
,	O	O
Ho	B-CITY	B-PROVINCE
Chi	I-CITY	I-PROVINCE
Minh	I-CITY	I-PROVINCE
City	I-CITY	O

Dictionary-based NER

✓ Advantages

- simple
- many readily available dictionaries/lexica

✗ Disadvantages

- dictionaries can become too big
- yet, none of them complete or comprehensive enough
- overlaps between categories, e.g., many people and places have the same names

Rule-based NER

- Regular expressions
  - checking for capitalisation
  - checking for numbers
- Function words for extracting, e.g., locations
  - Capitalized word + {city, centre, river} indicates location

    Examples: ***New York city, Hudson river***
  - Capitalized word + {street, boulevard, avenue} indicates location

    Examples: ***Fifth avenue***

Rule-based NER

- Context patterns
  - [*PERSON*] earned [*MONEY*]

    Example: *John earned £20*
  - [*PERSON*] joined [*ORGANISATION*]

    Example: *Sam joined IBM*
  - [*PERSON*], the [*JOBTITLE*]

    Example: *Mary, the teacher*

Rule-based NER

- still not so simple:

  [*PERSON|ORGANISATION*] fly to [*LOCATION*]

  Examples: *Jerry flew to Japan*

  Delta flies to Europe

  Birds fly to the nest

- match patterns defined in a gazetteer
  - dictionary of person names:

    [John, Jerry, Mary, Frank, David, …]

    Jerry is a person's name but not *Delta* nor *Birds*.

Rule-based NER

✓ Advantages

- handcrafted rules can be very precise
- only small amount of development data needed

✘ Disadvantages

- domain-dependent
- expensive development and test cycle

Shortcomings of dictionary- and rule-based approaches

- Failure to generalise
  - first word of a sentence is also usually capitalised
  - multiword expressions
- Inability to disambiguate
  - Jordan the **person** vs. Jordan the **location**
  - JFK the **person** vs. JFK the **airport**
  - May the **person** vs. May the **month**

Shortcomings of dictionary- and rule-based approaches

- Upkeep/maintenance
  - No gazetteer contains all existing proper names
  - New proper names constantly emerge
    - products, brands
    - scientific discoveries (e.g., planets, stars, medicines)
  - Multiple variants can emerge for the same entity
    - John Smith
    - J. Smith
    - Prof. J Smith

ML-based approaches to NER

- Supervised learning
  - labelled training examples
  - methods
    - hidden Markov models (HMMs)
    - naïve Bayes
    - decision trees
    - support vector machines (SVMs)
    - conditional random fields (CRFs)

ML-based approaches to NER

- Semi-supervised learning
  - small percentage of training examples is labelled, the rest is unlabelled
  - methods
    - bootstrapping
    - active learning
    - co-training
    - self-training
- Unsupervised learning
  - labels must be automatically discovered
  - method: clustering

Conditional random fields (CRFs)

- a widely used algorithm for sequence labelling
- finds the most probable label sequence **y** given an observation sequence **x**

$$\mathbf{y} = \operatorname{argmax}_{\mathbf{y}}\, p_\lambda(\mathbf{y}|\mathbf{x})$$

where **x** consists of the sequence of tokens from input text

Conditional random fields (CRFs)

- computation of probability

weight

feature function

$$p_\lambda(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \cdot \exp(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j(y_{i-1}, y_i, \mathbf{x}, i))$$

normalisation factor

summation over all tokens

summation over all feature functions

- feature function: characterises the input

$$f_i(x,y) = \begin{cases} 1, & \text{if } 1^{\text{st}} \text{ letter of } x \text{ is uppercase} \\ 0, & \text{otherwise} \end{cases}$$

Conditional random fields (CRFs): Feature types

- character *n*-grams (e.g., 2, 3, 4-grams)
- lexical and contextual
  - current word, lemma, part-of-speech (POS) tag
  - word *n*-grams: around W_0 in [-3,…,+3] window
- suffixes and prefixes (e.g., with lengths 2 to 4)

Conditional random fields (CRFs): Feature types

- orthographic

initial-caps	*all-caps*	*lonely-initial*
all-digits	*contains-dots*	*punctuation-mark*
single-char	*contains-hyphen*	

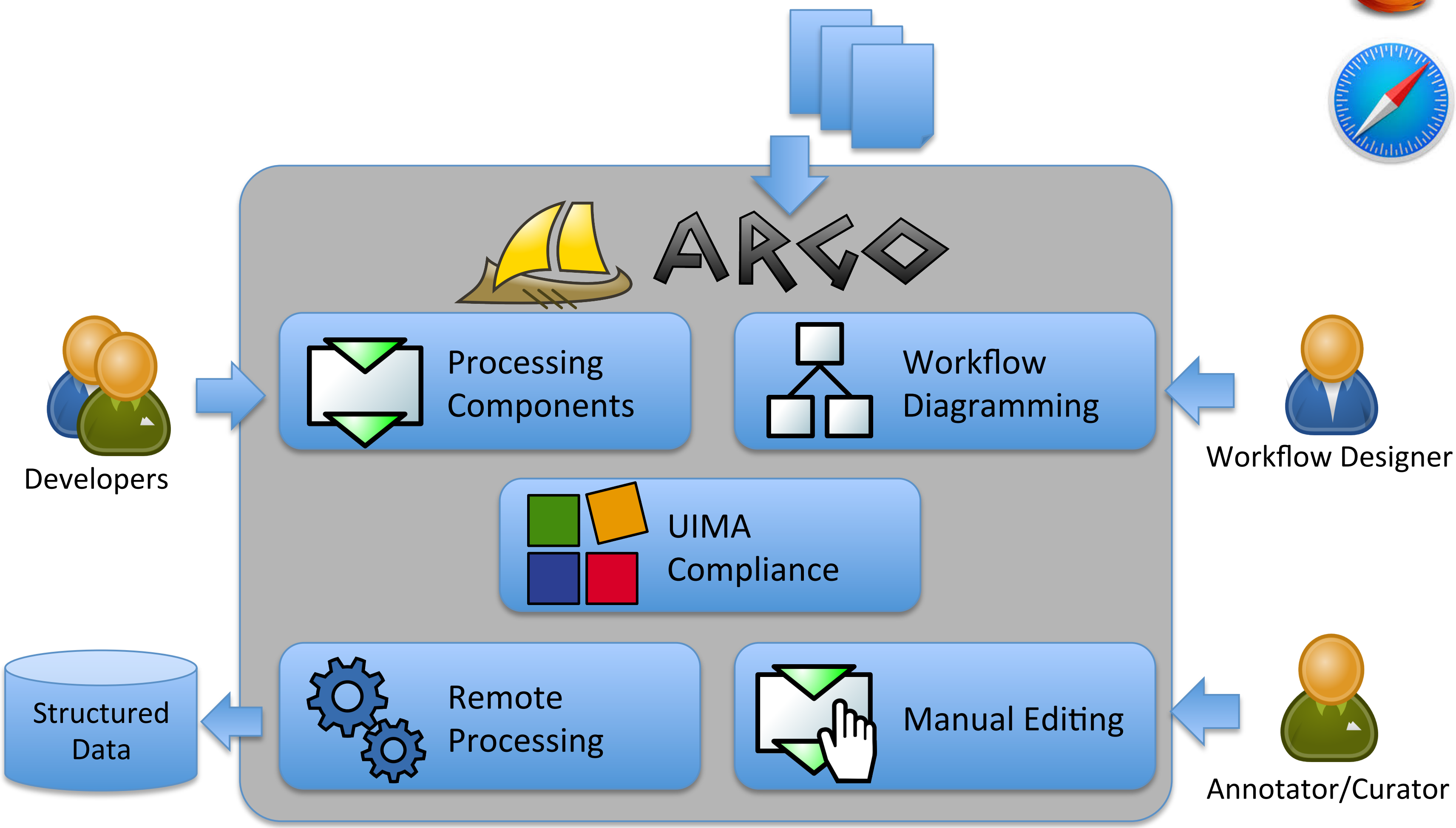
- semantic
  - matches between tokens and names in gazetteers or controlled vocabularies

Pipelining various tools for NER

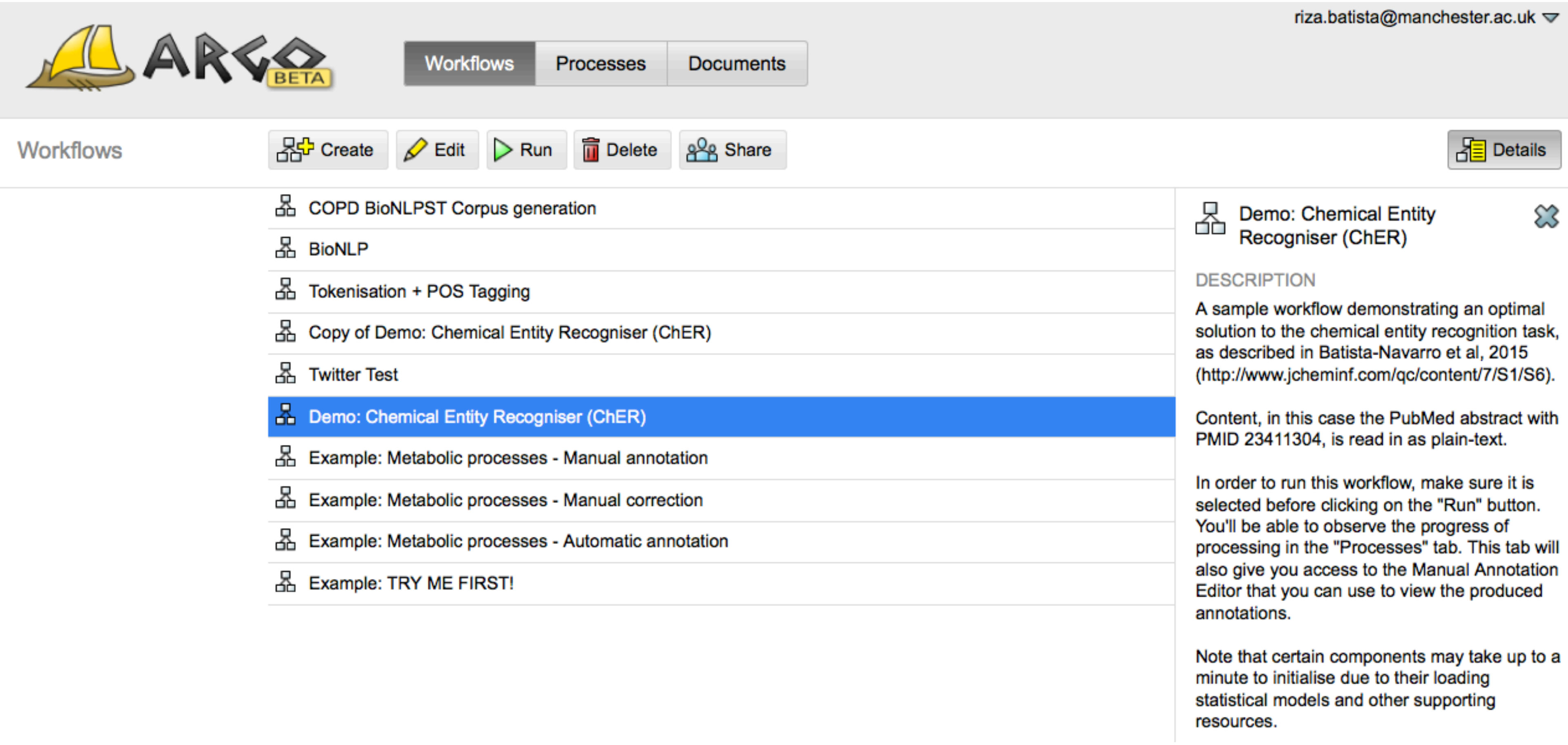
- Sentence splitting
  - to define a sequence
- Tokenisation
  - to generate the basic unit of analysis, i.e., tokens
- Lemmatisation, POS-tagging
  - to generate lexical and contextual features
- Gazetteer matching
  - to generate semantic features

Questions so far?

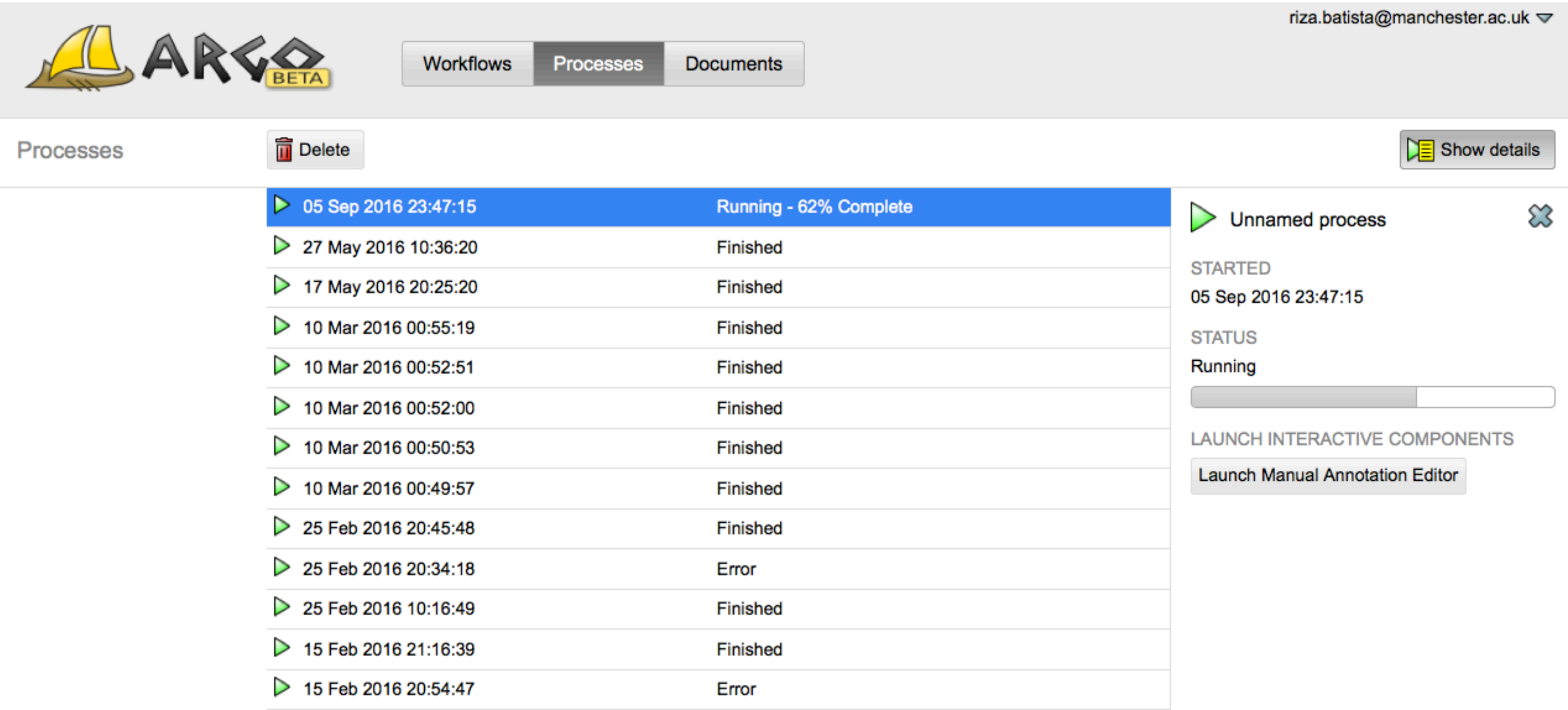
Argo: a generic text mining workbench (http://argo.nactem.ac.uk)



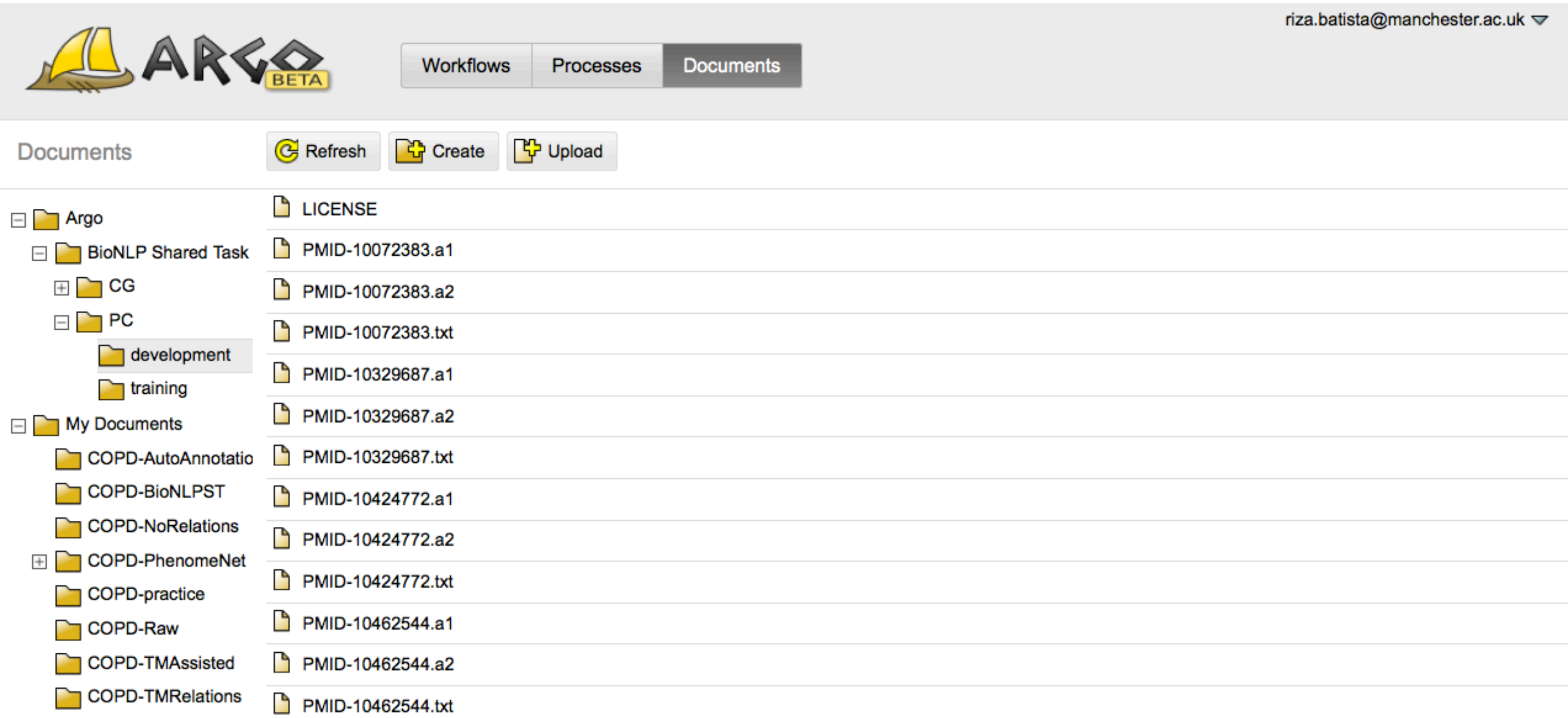
Workflows


riza.batista@manchester.ac.uk
ARGO BETA
Workflows
Processes
Documents
Workflows
Create
Edit
Run
Delete
Share
Details
COPD BioNLPST Corpus generation
BioNLP
Tokenisation + POS Tagging
Copy of Demo: Chemical Entity Recogniser (ChER)
Twitter Test
Demo: Chemical Entity Recogniser (ChER)
Example: Metabolic processes - Manual annotation
Example: Metabolic processes - Manual correction
Example: Metabolic processes - Automatic annotation
Example: TRY ME FIRST!
Demo: Chemical Entity Recogniser (ChER)
DESCRIPTION
A sample workflow demonstrating an optimal solution to the chemical entity recognition task, as described in Batista-Navarro et al, 2015 (http://www.jcheminf.com/qc/content/7/S1/S6).
Content, in this case the PubMed abstract with PMID 23411304, is read in as plain-text.
In order to run this workflow, make sure it is selected before clicking on the "Run" button. You'll be able to observe the progress of processing in the "Processes" tab. This tab will also give you access to the Manual Annotation Editor that you can use to view the produced annotations.
Note that certain components may take up to a minute to initialise due to their loading statistical models and other supporting resources.

Processes

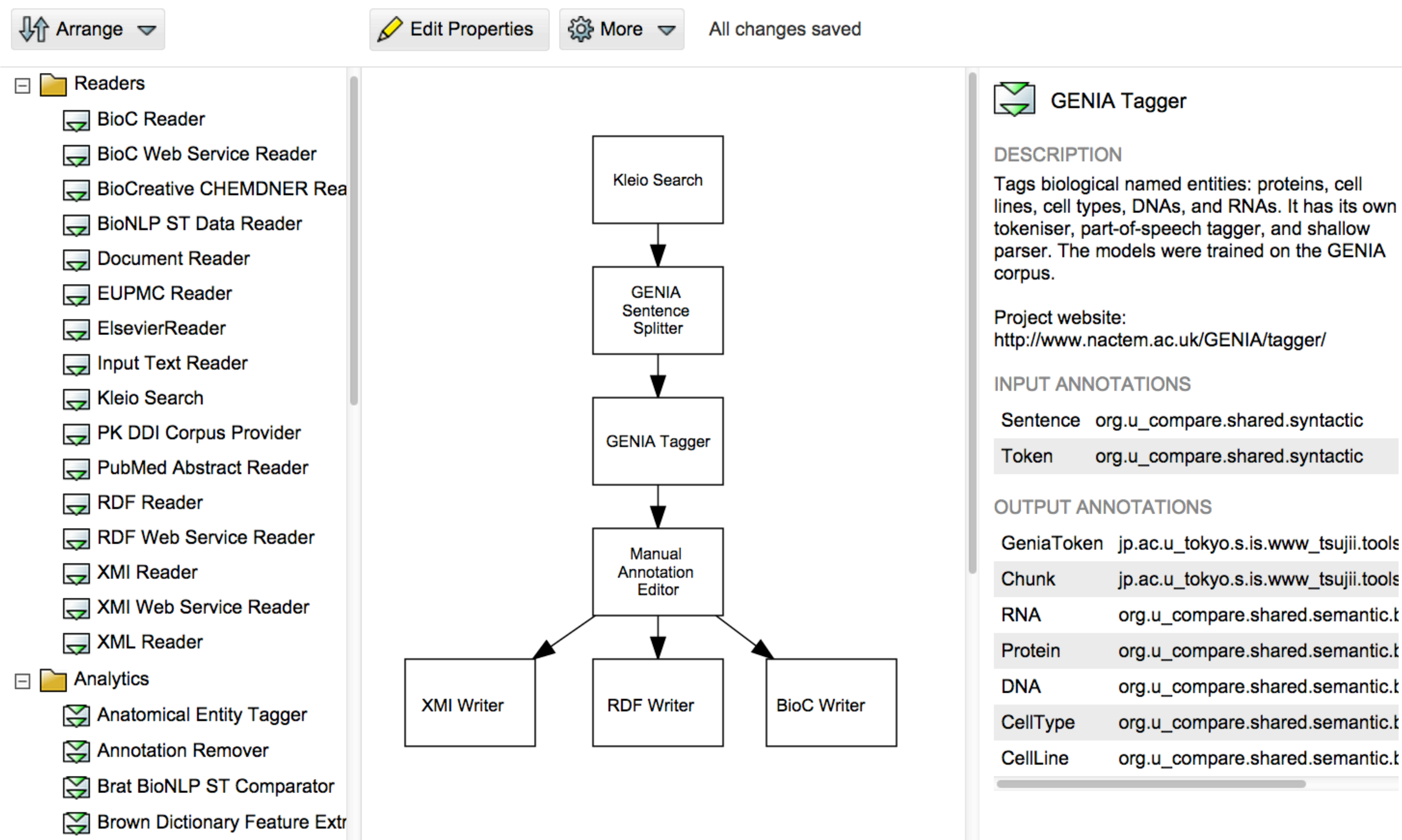

riza.batista@manchester.ac.uk
ARGO BETA
Workflows
Processes
Documents
Processes
Delete
Show details
05 Sep 2016 23:47:15 Running - 62% Complete
27 May 2016 10:36:20 Finished
17 May 2016 20:25:20 Finished
10 Mar 2016 00:55:19 Finished
10 Mar 2016 00:52:51 Finished
10 Mar 2016 00:52:00 Finished
10 Mar 2016 00:50:53 Finished
10 Mar 2016 00:49:57 Finished
25 Feb 2016 20:45:48 Finished
25 Feb 2016 20:34:18 Error
25 Feb 2016 10:16:49 Finished
15 Feb 2016 21:16:39 Finished
15 Feb 2016 20:54:47 Error
Unnamed process
STARTED
05 Sep 2016 23:47:15
STATUS
Running
LAUNCH INTERACTIVE COMPONENTS
Launch Manual Annotation Editor

Documents



ARGO
BETA
riza.batista@manchester.ac.uk
Workflows
Processes
Documents
Documents
Refresh
Create
Upload
Argo
BioNLP Shared Task
CG
PC
development
training
My Documents
COPD-AutoAnnotatio
COPD-BioNLPST
COPD-NoRelations
COPD-PhenomeNet
COPD-practice
COPD-Raw
COPD-TMAssisted
COPD-TMRelations
LICENSE
PMID-10072383.a1
PMID-10072383.a2
PMID-10072383.txt
PMID-10329687.a1
PMID-10329687.a2
PMID-10329687.txt
PMID-10424772.a1
PMID-10424772.a2
PMID-10424772.txt
PMID-10462544.a1
PMID-10462544.a2
PMID-10462544.txt

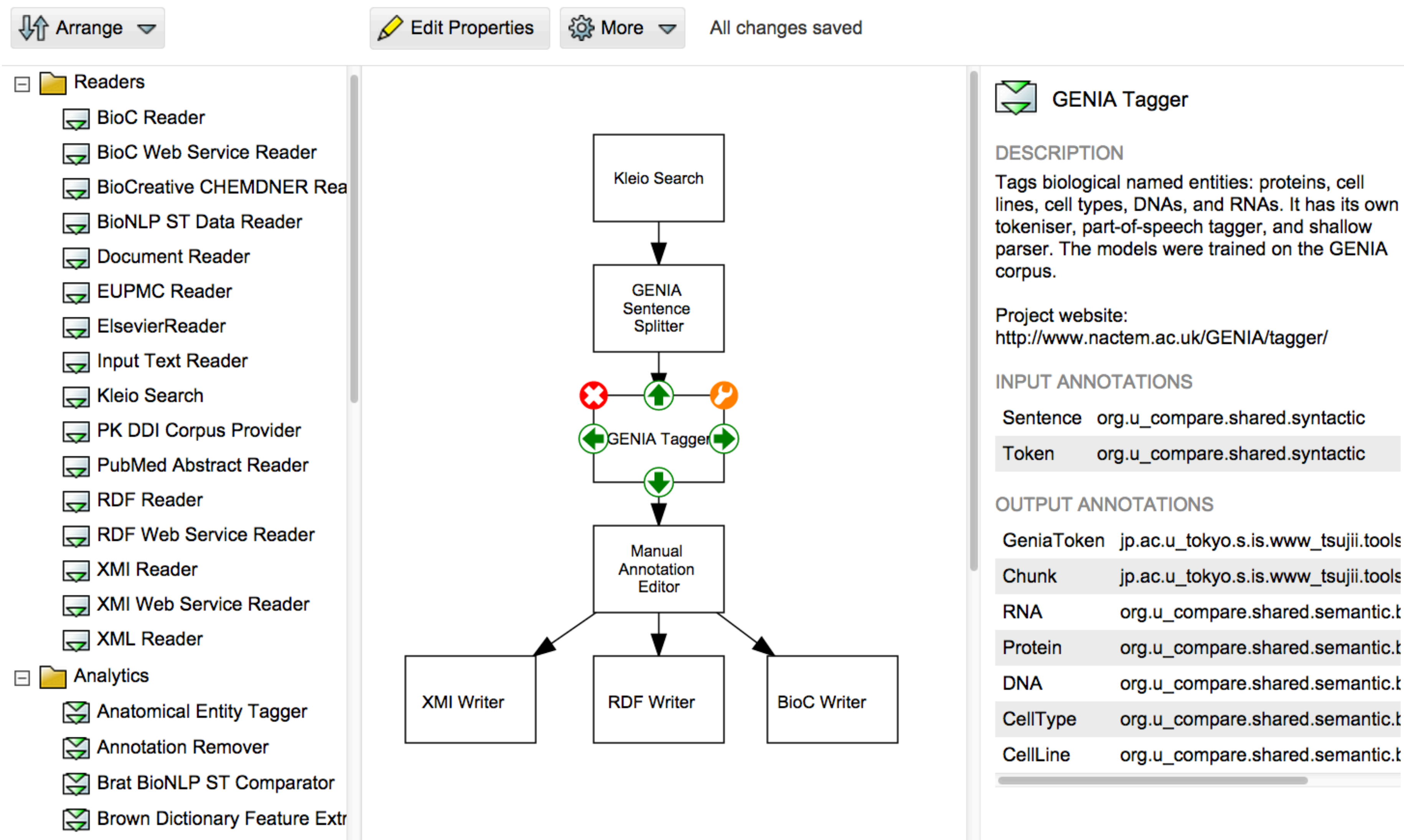
Workflow Editor



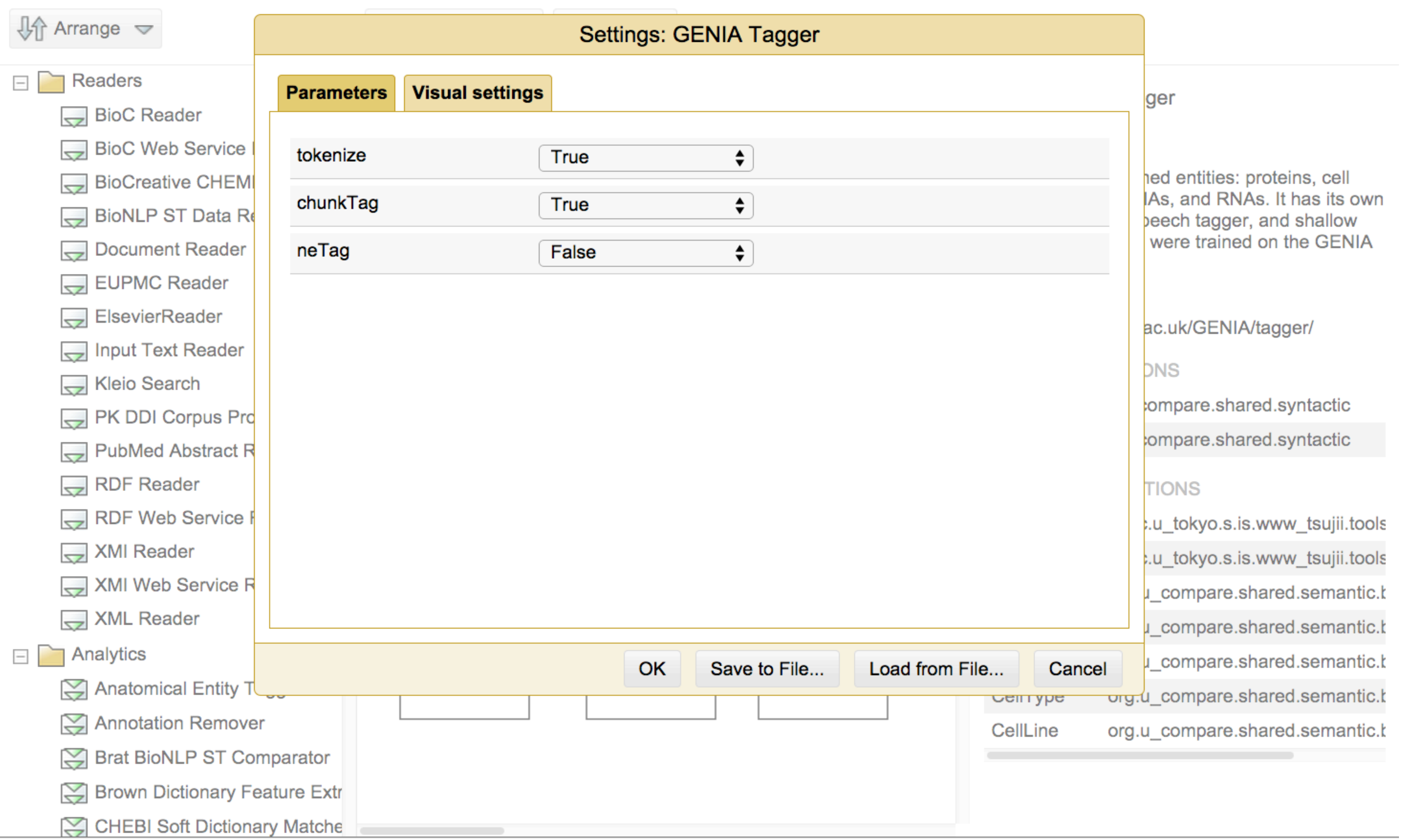
Components

- Readers
  - loads corpora/document collections
  - provide support for various formats, e.g., plain text, XML, TSV, stand-off
- Analytics
  - natural language processing tools
  - tokenisers, POS taggers, parsers, named entity recognisers
- Consumers
  - serialisation to files (e.g., XML, TSV) and databases

Configuration

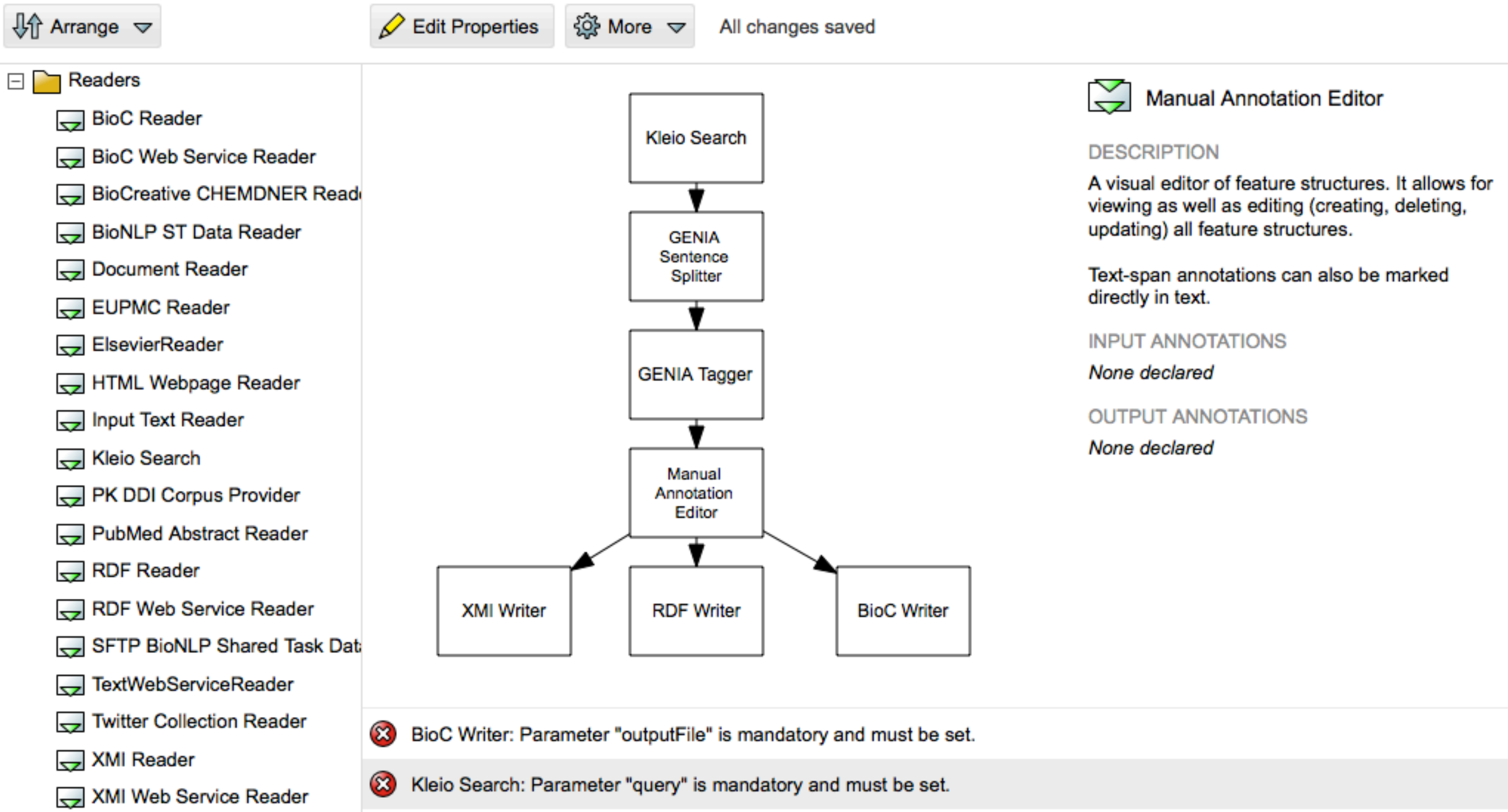


Configuration



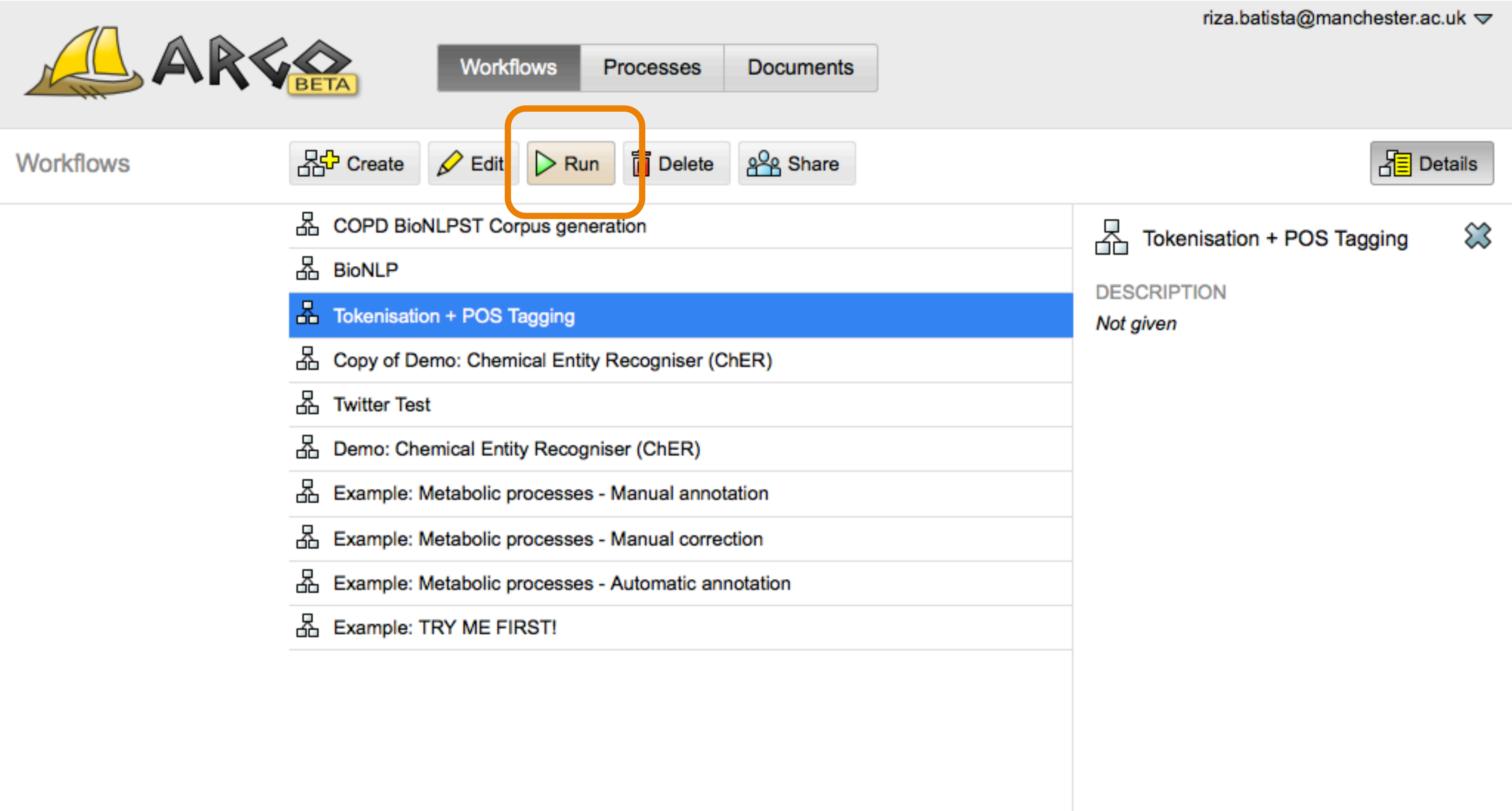
Argo: Workflow configuration

Execution



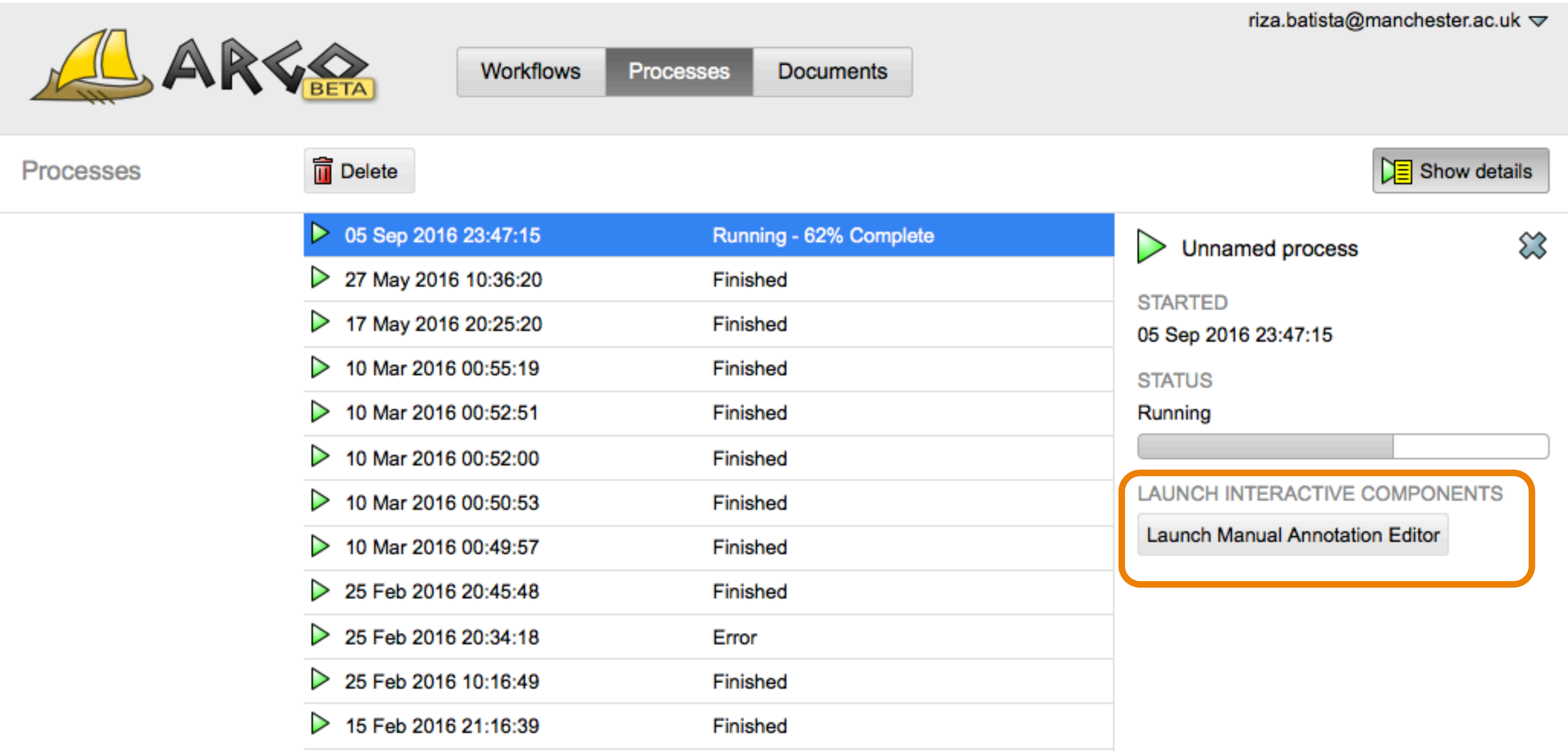
Argo: Workflow execution

Execution



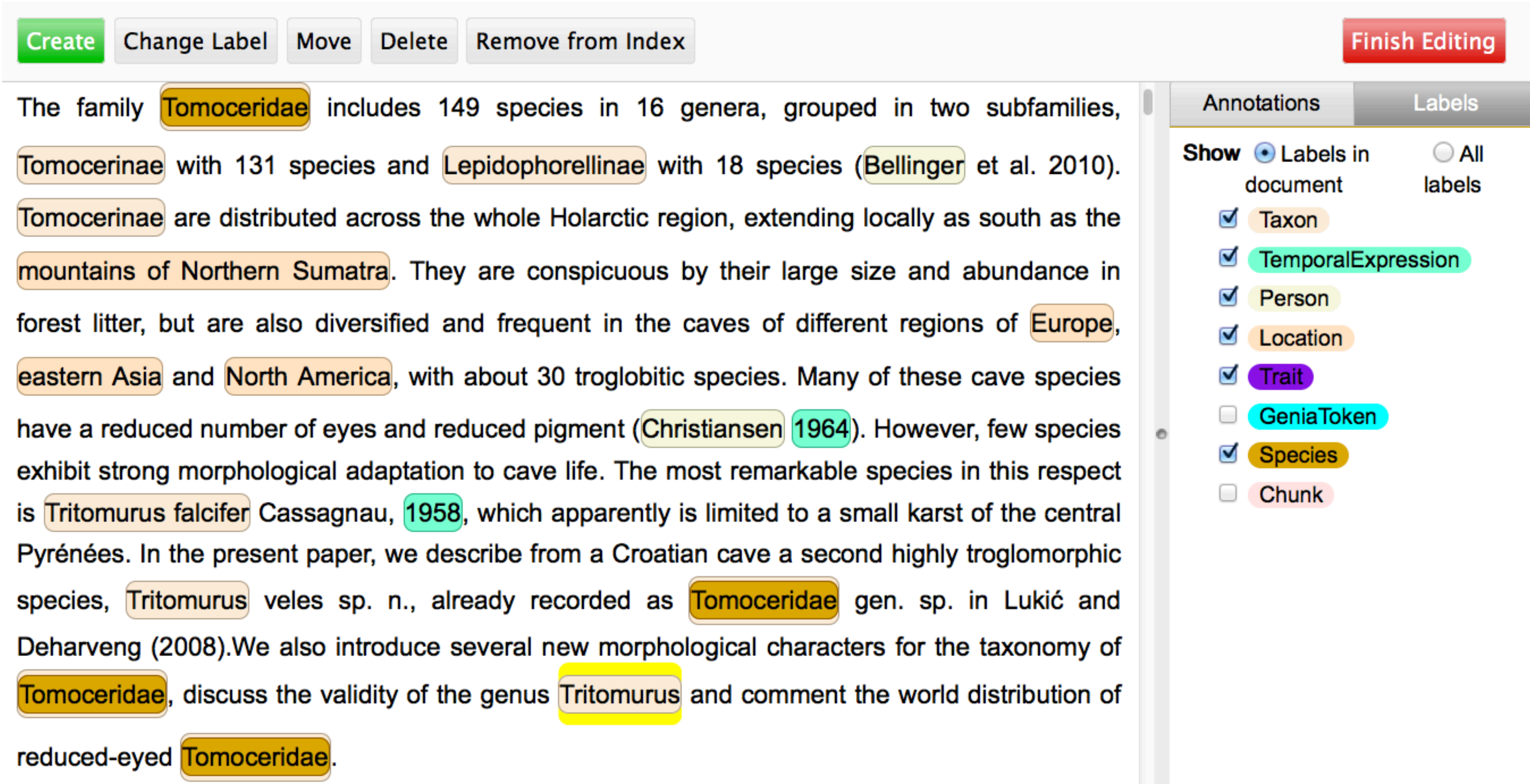
Argo: Workflow execution

Monitoring

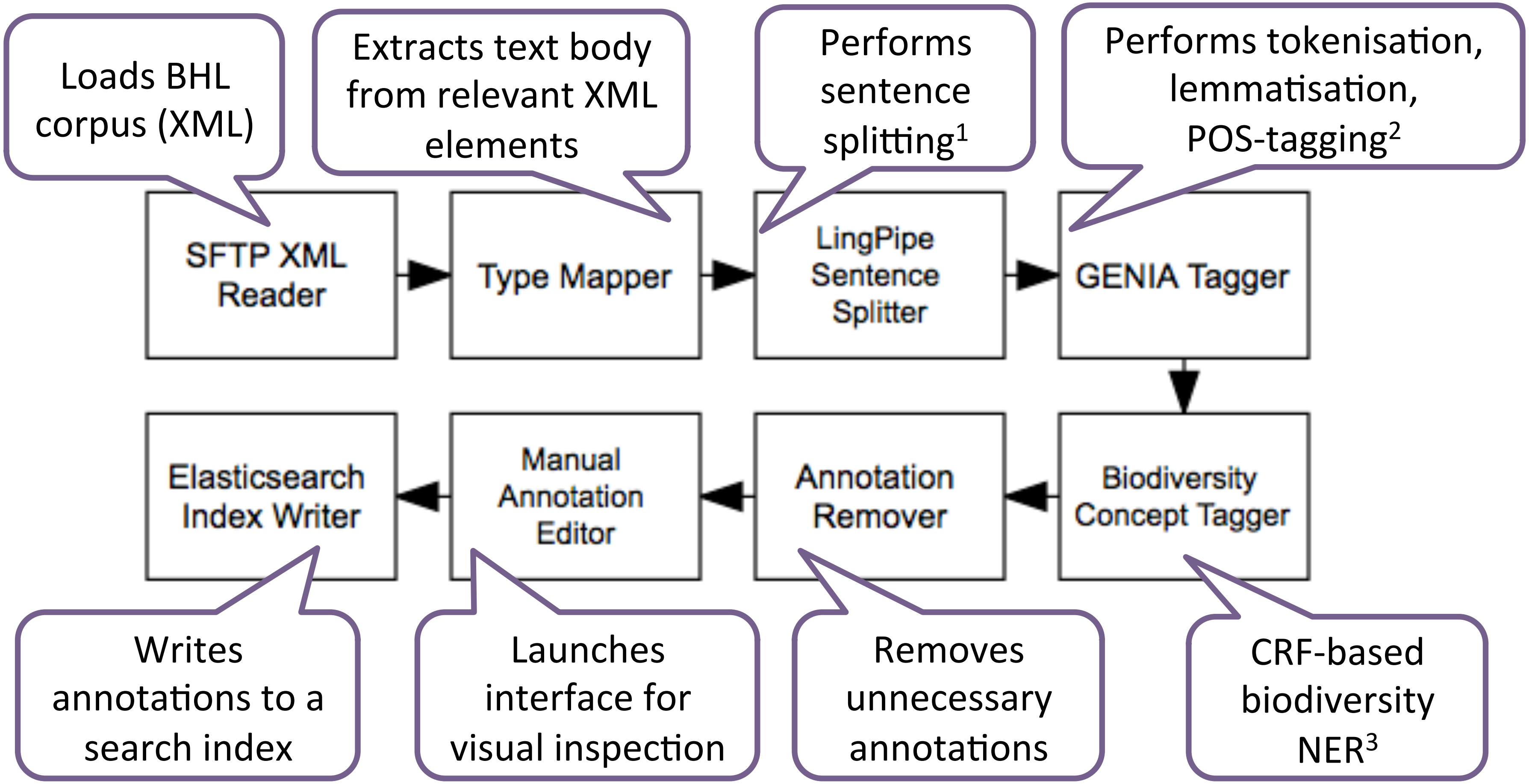


ARGO BETA
riza.batista@manchester.ac.uk
Workflows
Processes
Documents
Processes
Delete
Show details
05 Sep 2016 23:47:15 Running - 62% Complete
27 May 2016 10:36:20 Finished
17 May 2016 20:25:20 Finished
10 Mar 2016 00:55:19 Finished
10 Mar 2016 00:52:51 Finished
10 Mar 2016 00:52:00 Finished
10 Mar 2016 00:50:53 Finished
10 Mar 2016 00:49:57 Finished
25 Feb 2016 20:45:48 Finished
25 Feb 2016 20:34:18 Error
25 Feb 2016 10:16:49 Finished
15 Feb 2016 21:16:39 Finished
Unnamed process
STARTED
05 Sep 2016 23:47:15
STATUS
Running
LAUNCH INTERACTIVE COMPONENTS
Launch Manual Annotation Editor

Visual inspection of results: the Manual Annotation Editor

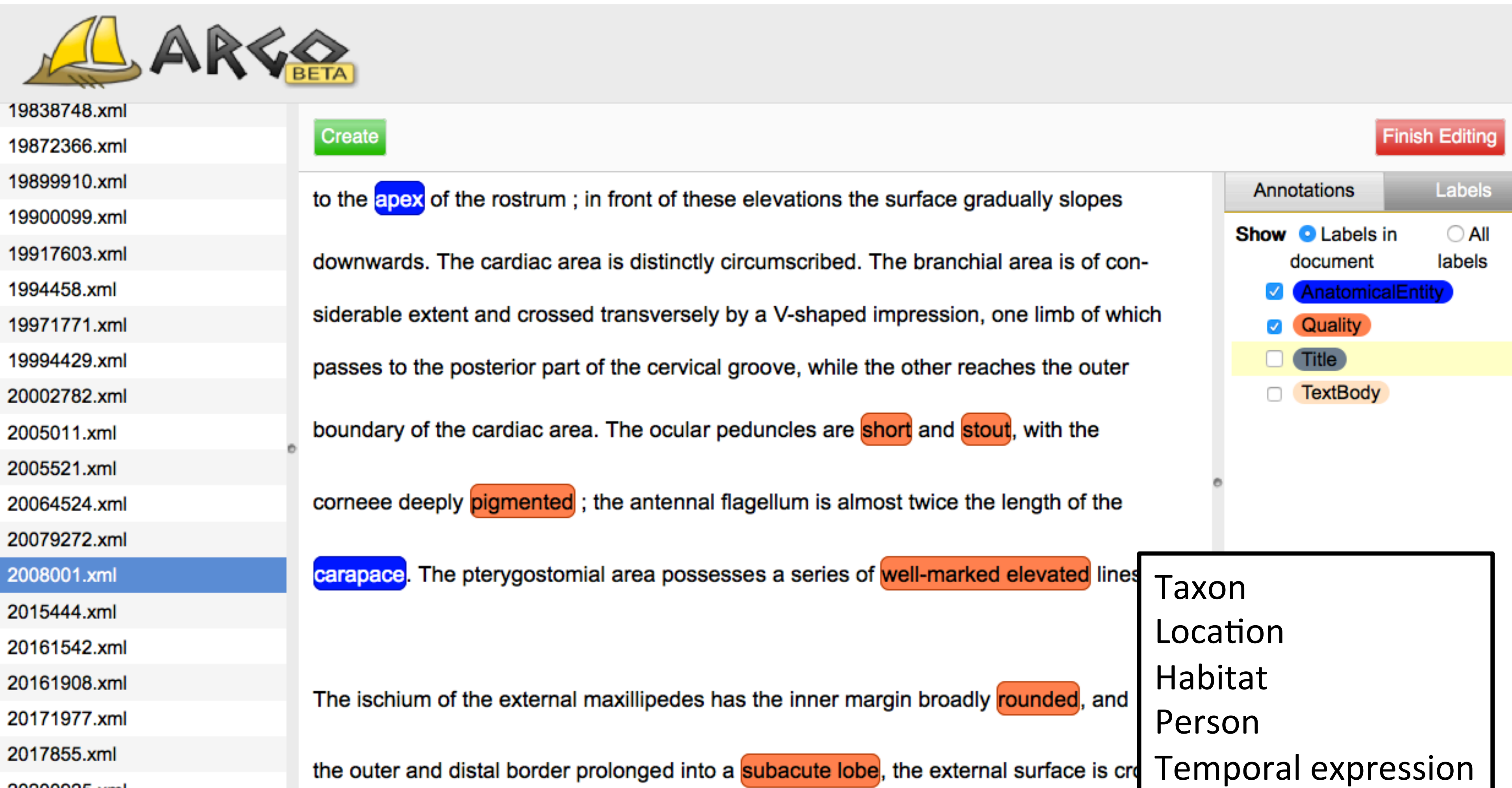


Generating semantic metadata with NER workflows: biodiversity literature



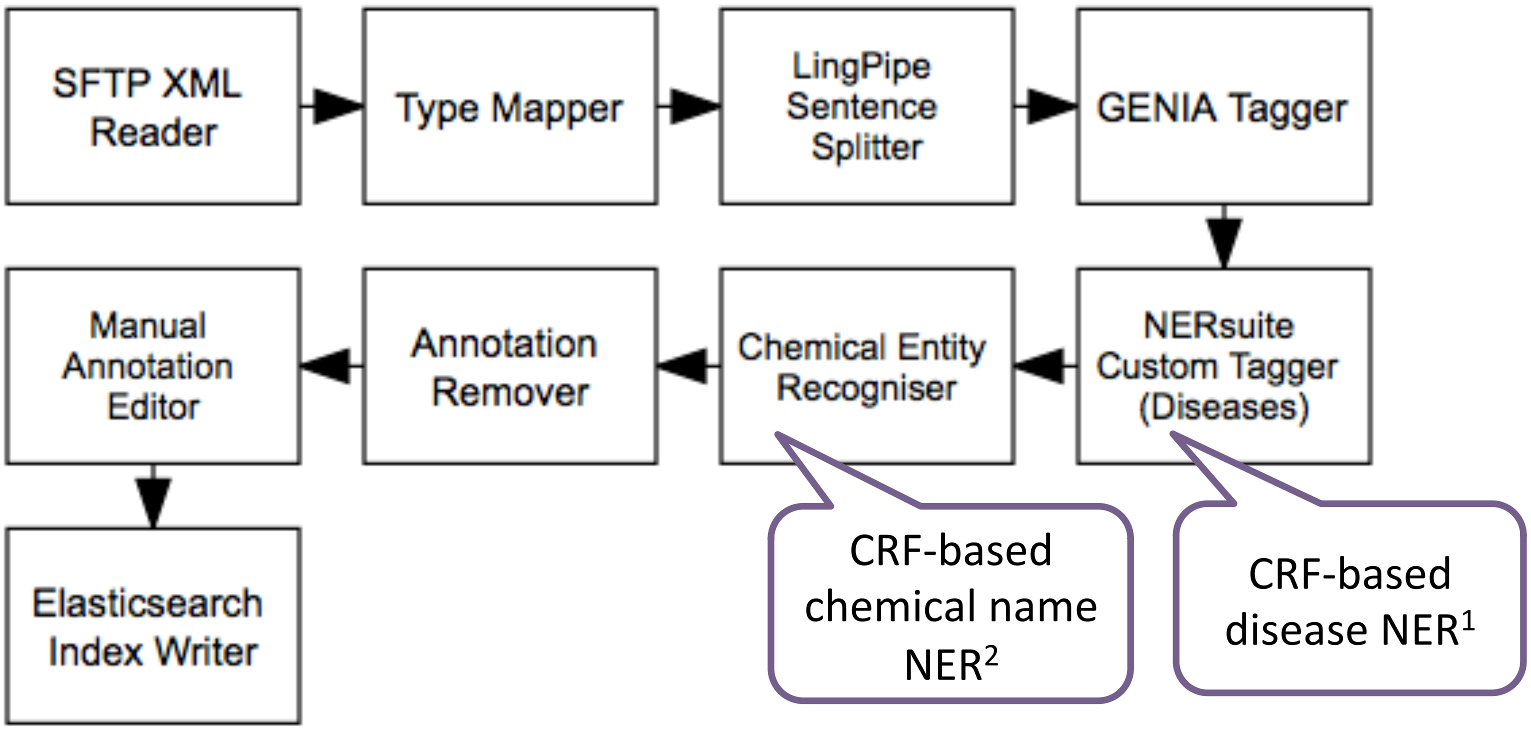
[1]LingPipe: http://alias-i.com/lingpipe
[2]GENIA Tagger: http://www.nactem.ac.uk/GENIA/tagger
[3]NERsuite: http://nersuite.nlplab.org

Generating semantic metadata with NER workflows: biodiversity literature



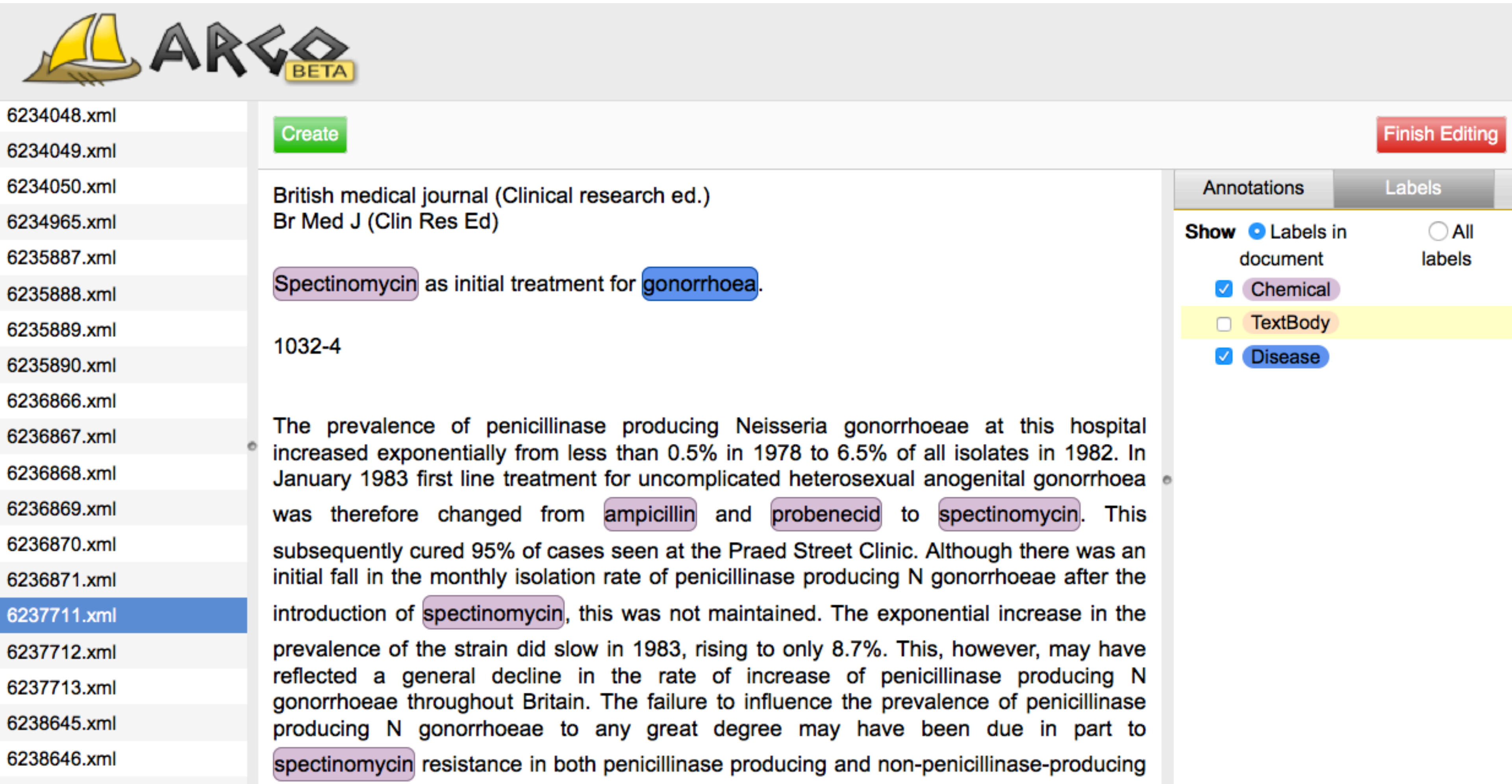
Taxon
Location
Habitat
Person
Temporal expression

Generating semantic metadata with NER workflows: medical archives



[1]NCBI Corpus: http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/DISEASE/
[2]ChER: https://jcheminf.springeropen.com/articles/10.1186/1758-2946-7-S1-S6

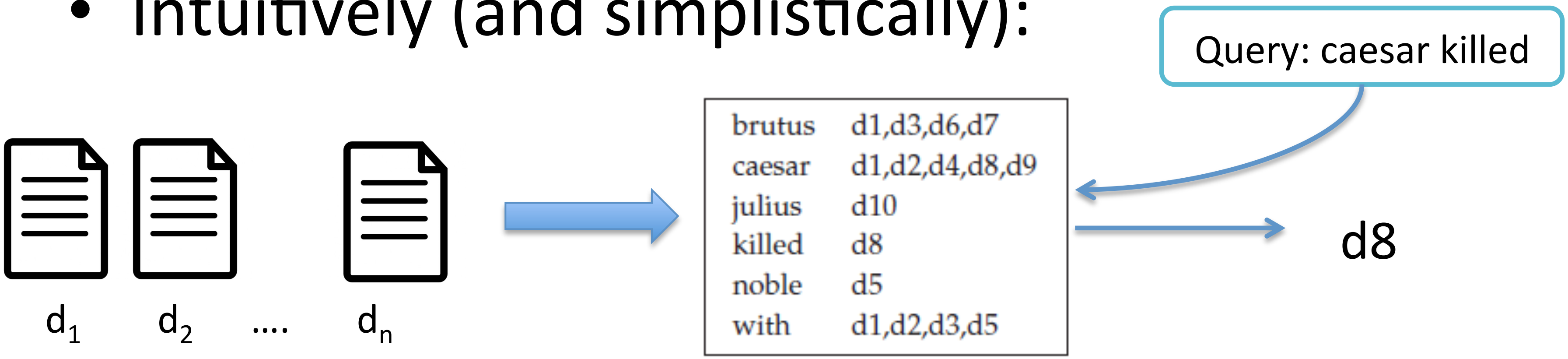
Generating semantic metadata with NER workflows: medical archives



Questions so far?

Introduction to Search Indices

- A **search engine** is an information retrieval system designed to help **find information stored** on a computer system
- Intuitively (and simplistically):



We will focus on Elasticsearch for this tutorial!

An overview of Elasticsearch[1]

- Elasticsearch is an open-source distributed search (full-text or structured) and analytics engine:
  - timestamp or exact values,
  - full-text search, handle synonyms, score documents by relevance
  - Analytics and aggregations from the same data in real time
- Notable examples:
  - Wikipedia (full-text search, highlighted snippets, and search-as-you-type and did-you-mean suggestions)
  - The Guardian (visitor logs with social-network data to provide analytics)
  - Stack Overflow (full-text search with geolocation queries and more-like-this in Q&A)
  - GitHub (query 130 billion lines of code)
- Elasticsearch can run on your laptop, or scale out to hundreds of servers and petabytes of data

[1]Much of the following content was extracted from the Elasticsearch documentation

An overview of Elasticsearch (cont)

- Built on top of Apache Lucene, a full-text search-engine library
- Lucene is arguably the most advanced, high-performance, and fully featured search engine
- Why not using Lucene then?
  - Complexity, requires a deep understanding of IR concepts and its inner workings
  - Need to work in Java and to integrate Lucene directly with your application
  - Elasticsearch packages up all this functionality into a standalone server that your application can talk to via (a RESTful) API
  - "Works right out of the box"; sensible defaults and hides complicated search theory, while still fully configurable and flexible

An overview of Elasticsearch

- Isn’t Solr doing the same?
  - Which one is better depends on the application
  - Elasticsearch was born in the age of REST APIs, so it’s more aligned with web 2.0 applications
  - In our case the nested document structure made Elasticsearch a clear winner
  - http://solr-vs-elasticsearch.com

How to install Elasticsearch

- It's quite straightforward:
  – https://www.elastic.co/guide/en/elasticsearch/guide/current/running-elasticsearch.html
- For development and interactive querying the recommended software is Sense
  – Available as a Chrome extension too
  – Send JSON data over HTTP
  – Friendly syntax for the curl command

How to communicate with Elasticsearch?

- Java API
  - Used within the Argo component
- RESTful API
  - Used for the examples here
- We will follow the 'learn from example' philosophy in this tutorial
  - Only emphasising important aspects of the query syntax

Elasticsearch key concepts

- Document oriented
  - Similar to the NoSQL concept of *document*
  - Intuitively, a document is analogous to an object in OO-programming
  - Why? No need to squeeze or flatten your object into a table (usually one field per column) losing its richness
- JSON
  - Serialisation format for documents

```
{
    "email":      "john@smith.com",
    "first_name": "John",
    "last_name":  "Smith",
    "info": {
        "bio":         "Eco-warrior and defender of the weak",
        "age":         25,
        "interests": [ "dolphins", "whales" ]
    },
    "join_date": "2014/05/01"
}
```

Elasticsearch key concepts

- Glossary:
  - Index:
    - analogous to a database in SQL and NoSQL
    - can contains multiple types
  - Type:
    - analogous to a table (SQL) or collection (MongoDB)
    - can contain multiple documents
  - Document:
    - analogous to a row (SQL)
    - can contain multiple fields
  - Field:
    - Analogous to a column (SQL)
    - Each field is associated with a field type: ‘string’, ‘date’, ‘integer’
- Index is an overloaded word
  - as a *noun*, as a *verb* and *inverted index*

Querying Elasticsearch

- We already ran Argo workflows, which inserted data in Elasticsearch
- Let's have a look at the existing indices…
- Let's search for all documents in an index…
  - Format of the response
  - Pagination

Querying Elasticsearch

- Let’s refine the query searching for a specific term…
- Let’s search for entities…

Querying Elasticsearch using Sense

```
GET /medical_archives/_search

GET /medical_history/_search

GET /medical_history/document/_search
{
  "size": 20,
  "query" : {
    "match": {
      "text": "AIDS"
    }
  }
}

# search for annotations!
GET /medical_history/document/_search
{
  "query" : {
    "nested": {
      "path": "metadata",
      "query": {
        "bool": {
          "must": [
            {
              "match": {
                "metadata.obj": "Disease"
              }
            },
            {
              "match": {
```


```
                "obj": "Disease",
                "word_form": "motor neurone disease"
              }
            ]
          }
        },
        {
          "_index": "medical_history",
          "_type": "document",
          "_id": "AVcG9YXp2vLZYWctVilx",
          "_score": 0.8829731,
          "_source": {
            "text": "\n                    A patient was admitted to hospital with an apparent psychiatric disturbance. When she became stuporous the cerebrospinal fluid was cultured but proved sterile. The latex test showed that serum was positive for cryptococcal antigens, and cryptococcal meningoencephalitis was diagnosed. Amphotericin B was given but when she developed a toxic reaction it was replaced by flucytosine. She responded well to flucytosine alone and no side effects appeared on continued treatment. Cryptococcal meningitis may present as a psychiatric disturbance, and serological tests are invaluable aids to diagnosis.\n            ",
            "metadata": [
              {
                "obj": "Source",
                "word_form": "1095135.xml"
              },
              {
                "span": {
                  "begin": 69,
                  "end": 92
                },
                "obj": "Disease",
                "word_form": "psychiatric disturbance"
              },
```

Some caveats

- No need to define a mapping (i.e. schema)
  - Elasticsearch tries to guess it ("works out of the box")
  - But in most cases it is necessary to define it:
    - Define nested objects as such (e.g. 'metadata')
    - Define fields that do not need text processing (e.g. metadata fields)
- Let's have a look at our current mappings…

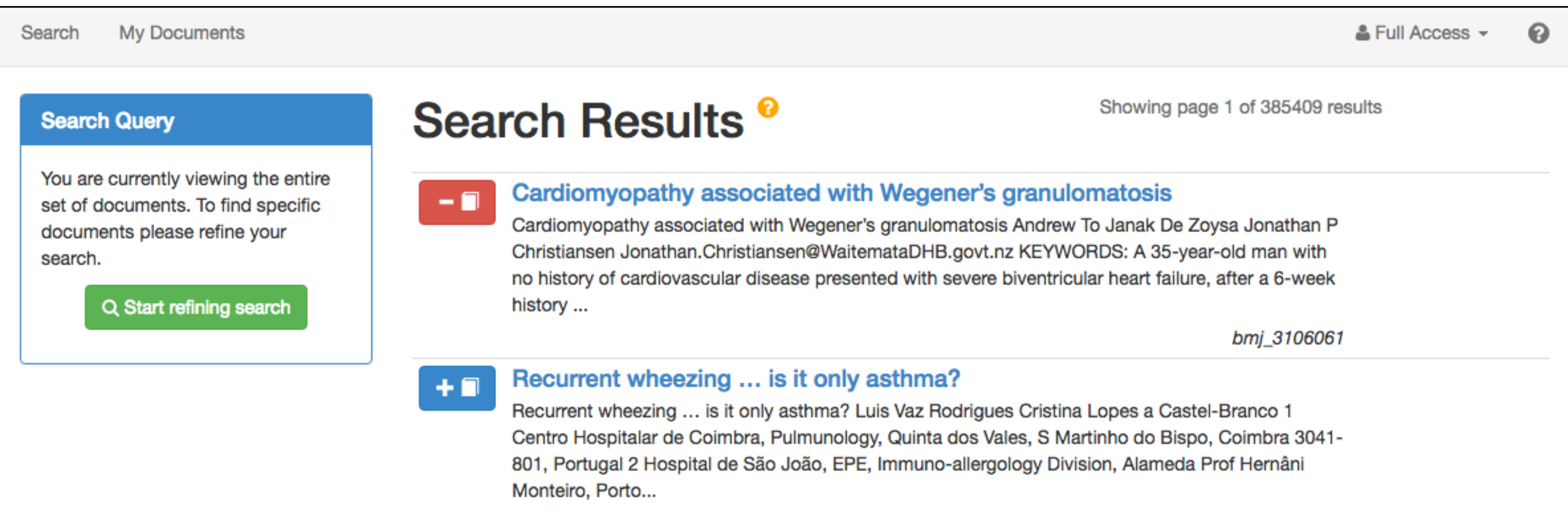
Much more…

- Aggregation (faceting)
- Horizontal scalability (sharding)
- Sorting / relevance
- Word proximity, partial matching, fuzzy matching, and language awareness
- Geolocation and geohashes

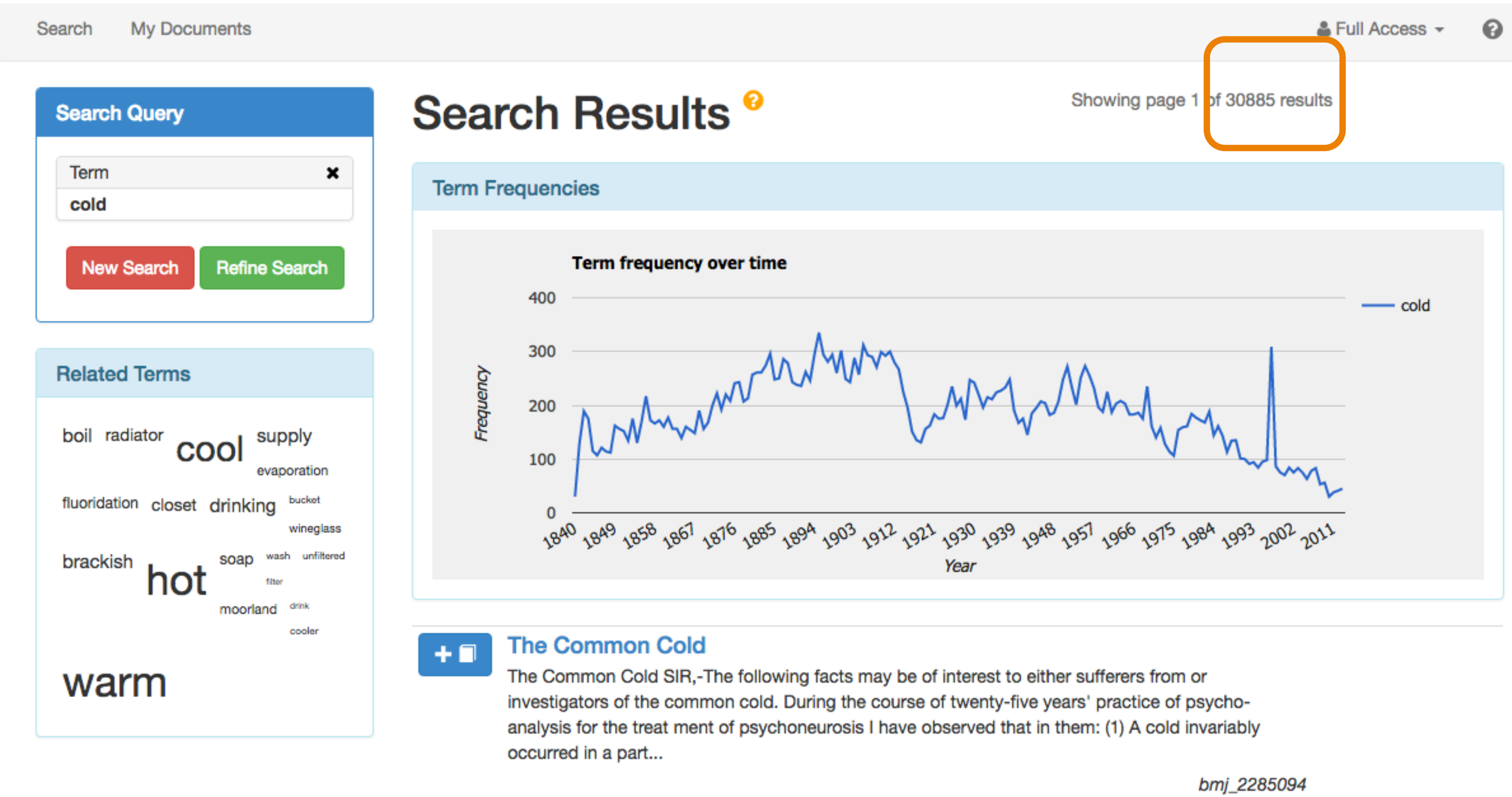
Questions so far?

Applications: Disambiguation in the History of Medicine search system

- http://nactem.ac.uk/hom
- Archives
  - British Medical Journal articles (380,000)
  - London Medical Office of Health reports (5,000)



Searching for “cold” based on keywords



Searching for "cold" based on keywords

"Cold" as a medical condition

The Common Cold

The Common Cold SIR,-The following facts may be of interest to either sufferers from or investigators of the common cold. During the course of twenty-five years' practice of psycho-analysis for the treat ment of psychoneurosis I have observed that in them: (1) A cold invariably occurred in a part...

bmj_2285094

POINTS FROM LETTERS: The Common Cold

POINTS FROM LETTERS The Common Cold Dr. E. WRIGLEY BRAITHWAITE (Leeds) writes: In 1943 (Journal, 2, 433) you published a letter from me in which I made a number of statements about the pathology of the common cold. . . . It only develops when the individual is in a particular emotional state, wh...

bmj_2039150

Why "A Cold"?

Why " A Cold "? SIR,-I have read Mr. James Crooks's article on nasal sin sinusitis in childhood (April 30, p. 935). In it the express tion "a cold," without any definition whatever, is repeatedly used. I suggest that the terms "a cold,"' ".catching cold," and " a chill " should be dropped from me...

bmj_2086387

"Cold" to describe temperature

Cold Drink and Syncope

770 BRITISH MEDICAL JOURNAL 23 Sen Ember 1972 Cold Drink and Syncope St,-Flight Lieutenant D. J. Rainford's letter (19 August, p. 475) reminds me of a verse I learnt at my mother's knee (she came from a medical family). ...

bmj_1788630

Feeling the Cold

Feeling the Cold SIR,-Your annotation on "Feeling the Cold" (February 21, p. 445) tells us that "cold

Searching for "cold" as a disease based on semantic metadata

Search My Documents Full Access

Search Query

Condition
cold

New Search Refine Search

Search Results

Showing page 1 of 3825 results

Diffuse pleural thickening and diaphragmatic paralysis causing combined intrathoracic and extrathora...

Diffuse pleural thickening and diaphragmatic paralysis causing combined intrathoracic and extrathoracic pulmonary restriction John J Dixon Grania J Price F Runa Ali 1 St. Helier Hospital, Nephrology, Wrythe Lane, Carshalten, London, SM5 1AA, UK 2 Guy's Hospital, Anaesthetics, Great Maze Pond, L...

bmj_3029080

Effect of point of care testing for C reactive protein and training in communication skills on antib...

Effect of point of care testing for C reactive protein and training in communication skills on antibiotic use in lower respiratory tract infections: cluster randomised trial Jochen W L Cals - general practitioner trainee and researcher Christopher C Butler - professor of primary care medicine Rog...

bmj_2677640

Monitoring the emergence of community transmission of influenza A/H1N1 2009 in England: a cross sect...

Monitoring the emergence of community transmission of influenza A/H1N1 2009 in England: a cross sectional opportunistic survey of self sampled telephone callers to NHS Direct Alex J Elliot - project lead Cassandra Powers - scientist Alicia Thornton - scientist Chinelo Obi - research assistant C...

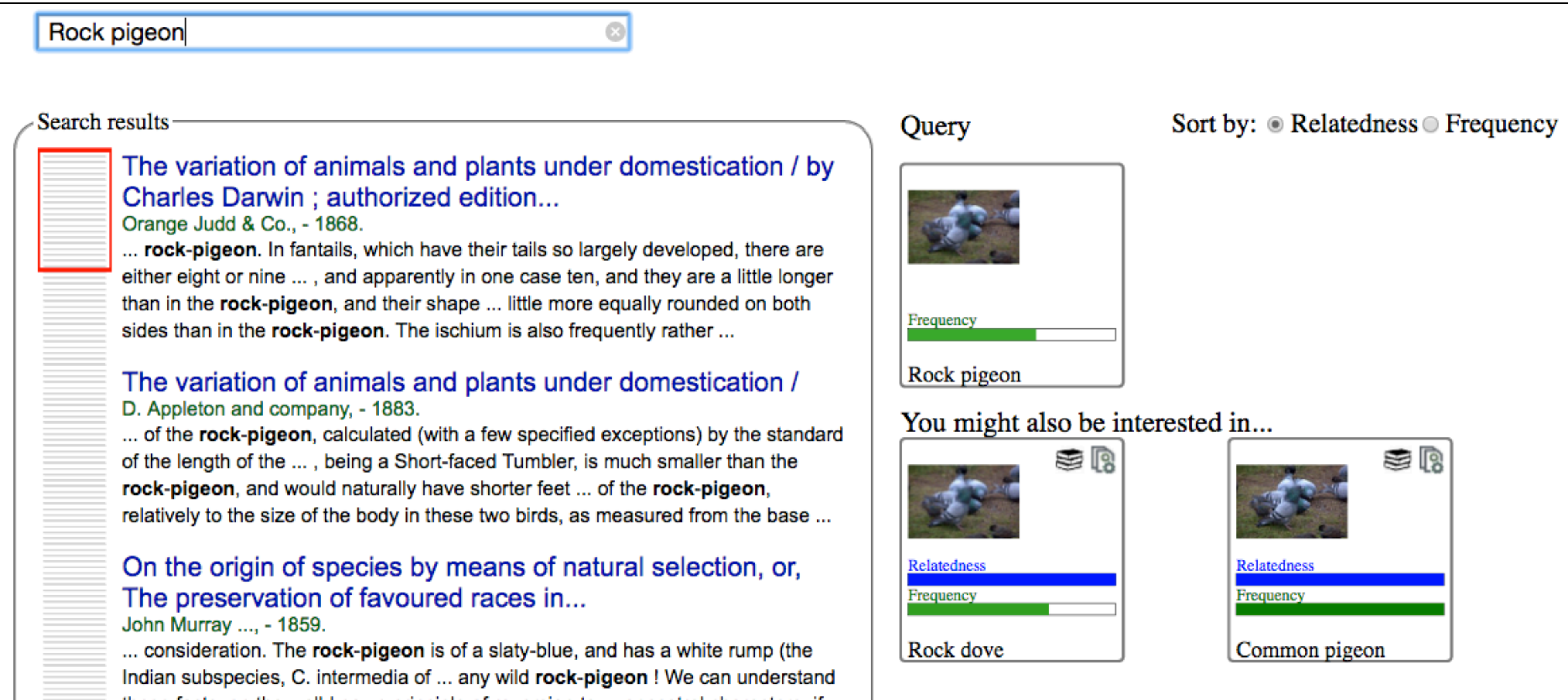
bmj_2733951

Effect of using an interactive booklet about childhood respiratory tract infections in primary care ...

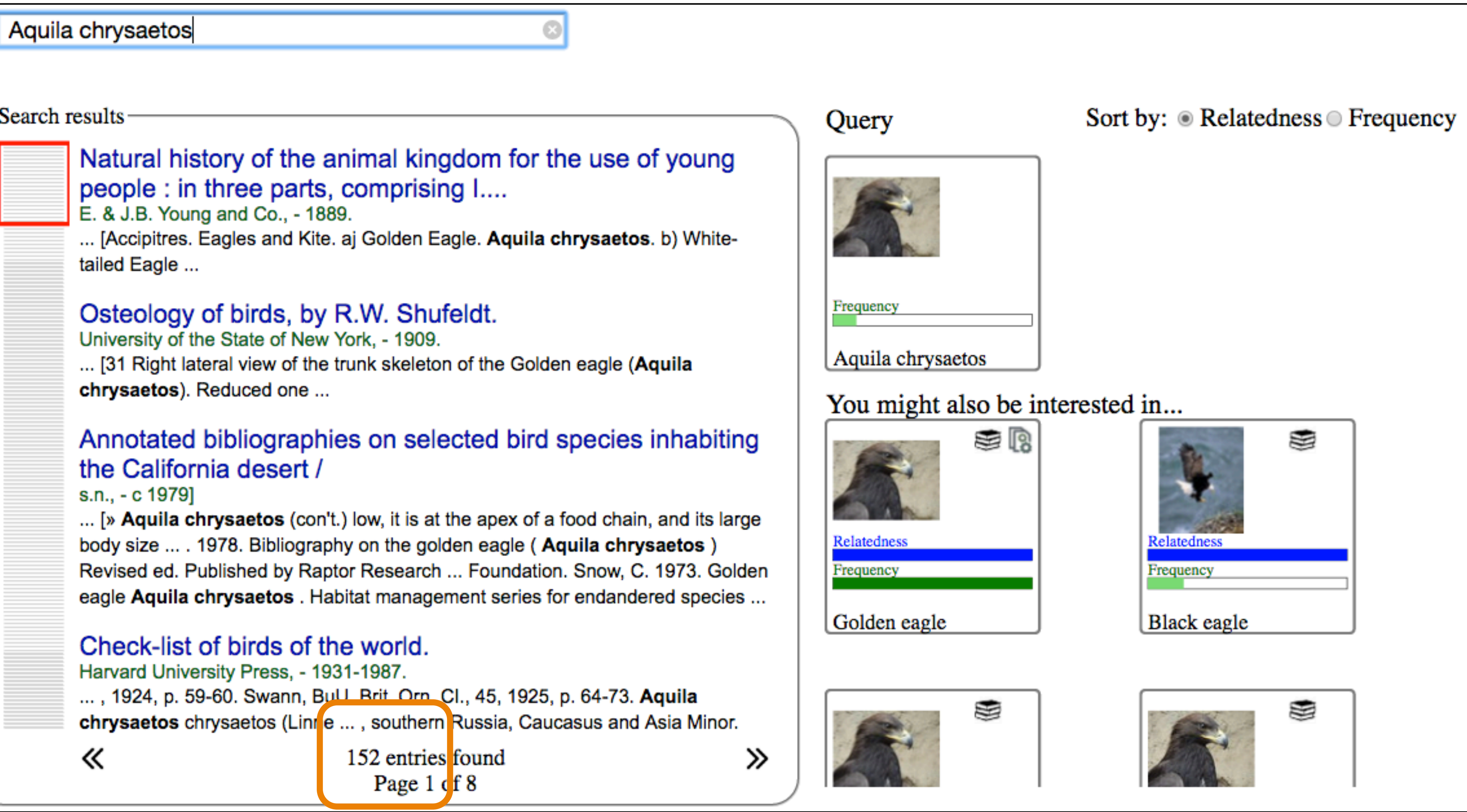
Effect of using an interactive booklet about childhood respiratory tract infections in primary care

Applications: BHL Query Expansion

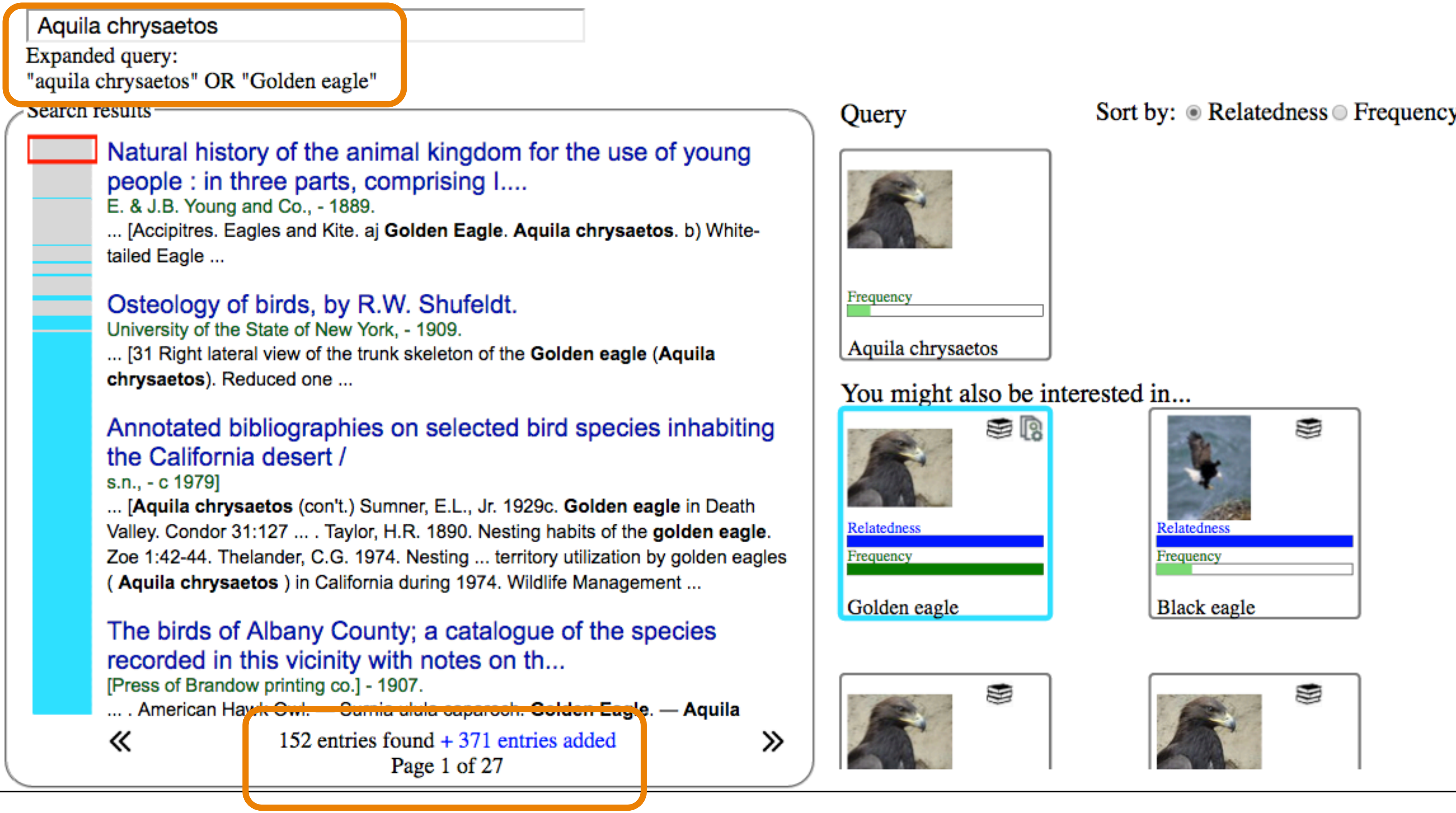
- http://nactem10.mib.man.ac.uk/va/MiBio/Search/queryExpansion.html?prot=thumb



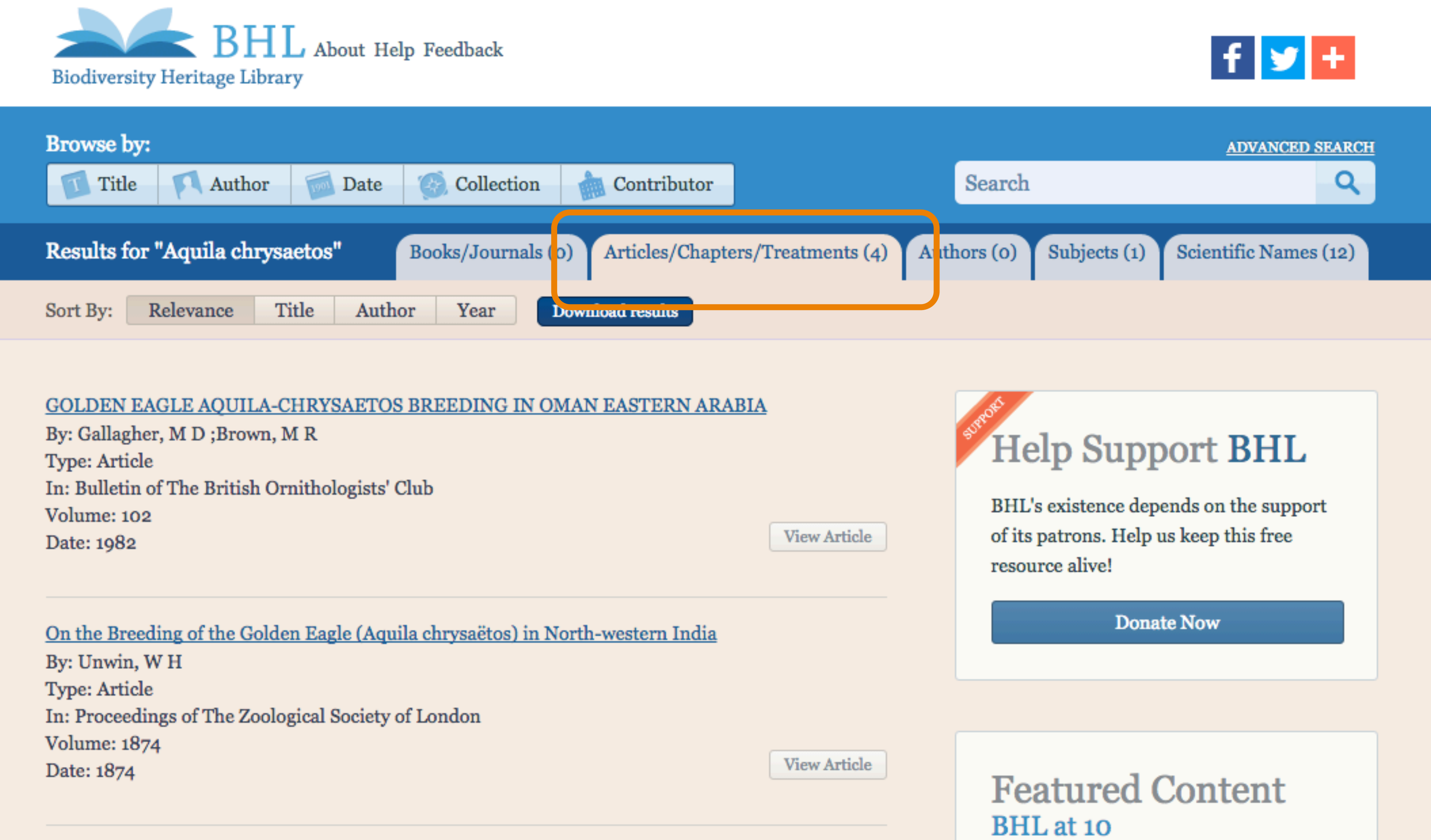
Searching for “Aquila chrysaetos”



Searching for “Aquila chrysaetos”: expanding with “Golden eagle”



Searching for "Aquila chrysaetos" in BHL



Conclusions

- Discussed challenges in information discovery and search
- Reviewed methods for NER
- Presented the Argo text mining workbench
- Extracted named entities which are then indexed to facilitate semantic searches
- Presented fundamentals of Elasticsearch: key concepts, search, mappings

Conclusions

- Illustrated some applications:
  - Disambiguation in the History of Medicine system
  - Improving recall in BHL
- Please get in touch with us if you're interested in applying Argo to your digital libraries!
  - riza.batista@manchester.ac.uk
  - axel.soto@manchester.ac.uk