Toolset for entity and semantic associations - Initial Release

Mark Lee, Petr Sojka, Radim Rehurek, Łukasz Bolikowski, Wojtek Hury, Volker Sorge, Thierry Bouche, Claude Goutorbe

EuDML
The EUROPEAN DIGITAL MATHEMATICS LIBRARY

ICTPSP
ICT POLICY SUPPORT PROGRAMME
part of the Competitiveness and Innovation Framework Programme CIP

DELIVERABLE

Project Acronym: EuDML

Grant Agreement number: 250503

Project Title: The European Digital Mathematics Library

D8.2: Toolset for Entity and Semantic Associations – Initial Release

Revision: 1.0 as of 27th May 2011

Authors:

Mark Lee	University of Birmingham, UB
Petr Sojka	Masaryk University, MU
Radim Řehůřek	Masaryk University, MU
Łukasz Bolikowski	University of Warsaw, ICM
Wojtek Hury	University of Warsaw, ICM
Volker Sorge	University of Birmingham, UB

Contributors:

Thierry Bouche, Claude Goutorbe CMD/UJF

Project co-funded by the European Comission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	✓
C	Confidential, only for members of the consortium and the Commission Services	

Revision History

Revision	Date	Author	Organisation	Description
0.1	24 November 2010	Petr Sojka	MU	First version as placeholder for partner's input.
0.2	18th February 2011	Petr Sojka	MU	gensim added as a tool.
0.3	3rd May 2011	Petr Sojka	MU	gensim demos added.
0.4	17th May 2011	Mark Lee	UB	Introduction added, some minor tweaking of English (incomplete).
0.5	19th May 2011	Mark Lee	UB	More proof reading and some pruning of text.
0.6	19th May 2011	Volker Sorge	UB	Added evaluation and example service in section 3. Rewrote introduction.
0.7	23rd May 2011	Petr Sojka	MU	gensim update (more docs).
0.8	25th May 2011	Volker Sorge	UB	Final edits.
0.9	26th May 2011	Thierry Bouche	CMD	Some changes to 3.4
1.0	27th May 2011	Volker Sorge	UB	Release for EU.

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Contents

Executive Summary
In this document we describe the initial release of the toolset for entity and semantic associations, integrating Unsupervised Document Clustering (initially implemented by MU) and Citation Indexing and Matching (as provided by ICM and UJF/CMD). We give a brief description of each tool and some initial evaluation.

1 Introduction

Mathematical works always build on previous results, and therefore can be considered a network of literature with rich links between individual documents. Sometimes such links are explicitly provided by metadata or can be inferred by subject classification (for example, MSC codes) but often such associations are either missing or incomplete. In addition, the corpus of mathematical literature is constantly growing which makes the ability to associate newly added documents with older documents in the collection highly desirable. Consequently we develop and integrate tools in EuDML that allow us to automatically set up a network structure associating related works within a collection of documents. This report describes the initial version of the toolset for automatic semantic association between documents within and outside of the EuDML collection.

Our review of the state of the art in D8.1 [5] identified two technologies which are mature enough to facilitate the discovery of such associations.

Citation Indexing aims to create a network of documents within the collection by automatic parsing and linking of citations.

Similarity Clustering tries to collate articles in collection with similar topic and content. It can be achieved either by automatically classifying documents according to a pre-defined scheme (e.g. MSC) or clustering documents based on co-occurring terms.

Partners in this work package have extensive experience in both these areas and have already developed relevant technologies and therefore it was decided that the initial prototype for this workpackage would consist of a suite of APIs for partner's existing technology including:

- MU's work on Unsupervised Document Clustering;
- ICM's work on Citation Indexing and Matching;
- UJF/CMD's work on Citation Indexing and Matching.

This current document describes this toolset and provides a brief description of each tool and some initial evaluation. In Section 2 we describe document classification and clustering and in Section 3 we describe tools for citation indexing and matching. Finally in Section 4, we discuss some initial evaluation of the toolset and future work.

2 Similarity/Clustering Services and Tools

The key functionality of these services is finding documents similar to a given document. The expected set should consist of documents which are semantically related—but which features are used during similarity search depends on the particular implementation.

The service basically has two operations:

- **Indexing** – All documents must be 'indexed' before being made available to similarity search. Indexing is done only once for each document and so it can be considered as a setup process for the service. The method of indexing may vary between implementations but one requirement is the facility for incremental indexing where new documents can be added to the collection and indexed without the need to re-index the entire collection.
- **Similarity search** – the core functionality of finding similar documents to particular document. Given a set of documents which are indexed (i.e. the collection), the similarity service should return a list of documents which are similar to specified one. It should be noted that different similarity metrics may be defined (in fact each implementation of the service is a realization of a particular metric) and so comparison of different implementations heavily depends on the chosen similarity metrics.

In the next two subsections we will focus on the available implementations: gensim library and Yadda similarity service implementation.

2.1 Gensim

The goal of this tool is to assist humans in data exploration via similarity browsing. GENSIM [4] is a software framework for modelling semantic similarity of text documents. Within the context of digital libraries, it allows querying for "similar documents", with similarity based purely on document contents (i.e. plain text, not metadata).

2.1.1 Gensim Demo

To demonstrate the capabilities of GENSIM, an Internet-based demo has been produced which showcases four possible scenarios (similarity demos):

1. For a given article, what are its ten most similar articles in the library?
2. Given two articles, how similar are they?
3. What are the pairs of the most similar articles across the entire collection (plagiarism candidates)?
4. Given an article, what are the topics covered by this article (data exploration)?

The demo can be found at `http://nlp.fi.muni.cz/projekty/eudml/gensim/`.

As not enough EuDML data was available at the time of the demo preparation, papers from the MREC collection of mathematical texts (a snapshot of 434,894 full-text articles from ARXMLIV, originally from ARXIV) were used to prepare the demonstration and to verify the framework's scalability.

2.1.2 Statistical Semantics

GENSIM contains several automated algorithms for deriving semantic representation from plain text, and therefore a choice is given to demo users between Latent Dirichlet Allocation (LDA) [1], Latent Semantic Analysis (LSA) [3] and plain TF-IDF in Demos 1–3, by means of a drop-down list. The first two methods compute a higher-level, semantic similarity, the last one only measures (weighted) word overlap.

Semantic properties of LSA and LDA come from exploiting word co-occurrence within documents. In a training corpus of documents, words that tend to appear together

are taken to be semantically related and soft-clustered together ("soft" because a word belongs to each cluster with a weight, or probability, not as a binary decision). Each such cluster of words describes one topic. Obtaining the clustering automatically and efficiently is a major challenge; GENSIM is a leading framework in scalable model training.

Given a trained LSA or LDA model, any text document can be described by how much it belongs to each topic. This gives a higher level (more abstract) representation of the document's contents—now even two documents that do not share any words in common can be evaluated as closely similar. This is a strict departure from an exact keyword overlap as popularized in search engines with boolean keyword searches.

2.1.3 Gensim Integration

This demo is a stand-alone application. Integration of GENSIM into the existing EuDML similarity interfaces (similarity Demo 1 and 2), with EuDML data instead of MREC, is in progress. EuDML currently does not contain interfaces for the tasks offered in similarity Demos 3 and 4.

GENSIM does not require any tagged metadata, such as an article's category, language, authors, citations etc. Extensions to include mathematics mark-up in plain text are possible, pending evaluation of their usefulness on real EuDML data.

2.2 Yadda Similarity Service

The Yadda similarity service is implemented over the Lucene full text search engine [6] and its "more like this" search functionality. Indexing consists of building an inverted index of documents, i.e. a mapping between words (terms) and the documents which contain them together with some additional statistics (for instance the number of occurrences in each document). The inverted index is used for similarity searches in the following fashion:

- A given document's important features are determined during search. For full text search, any 'feature' is just a word which can be considered specific or highly significant for the given document. Such words are chosen using term frequency/inverse document frequency (TF-IDF) statistics for each word.
- The most specific/significant words of the document are used to construct a boolean OR full text query—standard full text search and its "term vector model" is used to determine set of documents which match the query in the best way (documents which are most similar to the words in the query). Documents found during full-text search are returned as similarity results.

There is one specific feature of the Yadda similarity service—documents are not stored in one single index but are separated in multiple language indexes. The Yadda language detection algorithm is used during indexing—such an approach is sensible, because full-text similarity search is based on words frequencies and so should be applied to documents in the same language.

To summarise Yadda similarity service implementation we list pros and cons of the solution:

- **Pros**
  - Incremental indexing is available
  - Similarity search is fast

  - Query-time features detection allows to ask about documents similar not only to already indexed document but also to ad-hoc given document
- **Cons**
  - The algorithm works reasonably well only for document written in the same language. It should be noted that Yadda automatic language detection algorithm partly addresses that problem (documents in different languages are separated).
  - The algorithm uses just TF-IDF feature detection and term vector model search. No advanced similarity algorithms are used.

3 Linking and Matching Tools

Drawing links between articles in a collection is important to support the rapid detection and retrieval of articles similar to a given input article or query and to serve the results to the user. Since many documents in the EuDML collection are likely to contain bibliographic references, and many of these references are likely to point to (other) documents within the EuDML collection we are aiming to use *Bibliographic Reference Matching* as the primary means to build up a relationship network between articles. The goal of bibliographic reference matching is to assign to a bibliographic reference an identifier of the referenced document.

3.1 Idea

To achieve this goal, each document and each of its bibliographic references are indexed upon addition to the metadata storage. Following that, for each added document we are looking for:

- bibliographic references matching the document being added;
- documents matching the bibliographic references of the document being added.

In this mode of operation, it is possible to match the incoming documents on-the-fly, as they are incrementally added to the metadata storage. There is no need to recalculate the entire collection after a small update of the storage, such as adding or removing a couple of documents. The faster updates come at a cost, though. The metadata index must now contain not only the documents, but also bibliographic references, so it grows in size by approximately an order of magnitude.

3.2 Implementation

Bibliographic reference matching is implemented as a set of *processing nodes*. One node creates document metadata for each bibliographic reference. Another one matches a given document and its bibliographic references with relevant entities in the index. Yet another node updates NLM entries with matched document identifiers.

Matching of a document or bibliographic reference is performed by a series of queries to the metadata index. Firstly, if we know an identifier of the entity being matched, such as DOI, MR or Zbl, we query the index for documents with one of these identifiers. Otherwise, we query the index for documents given authors' surnames, a hash function

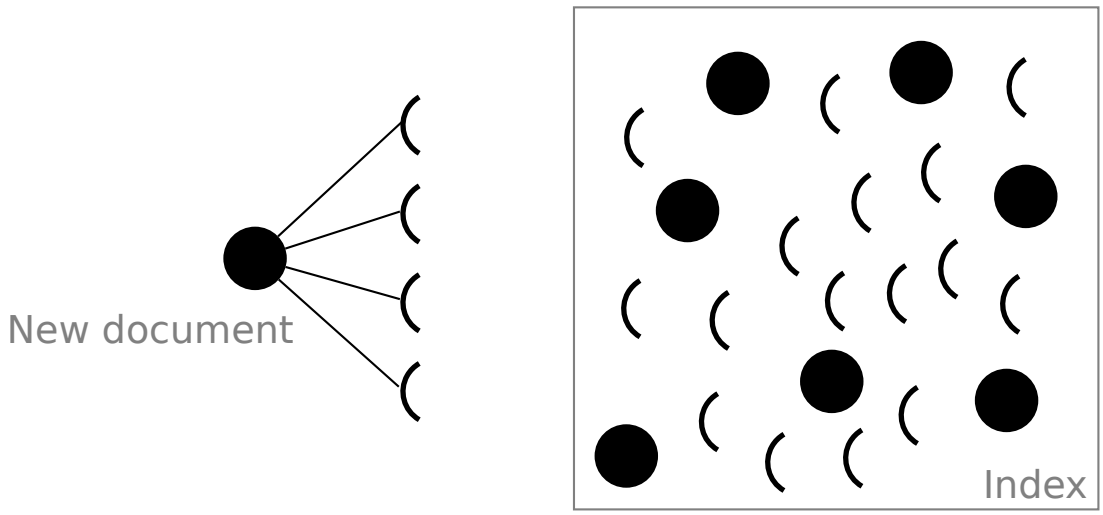

Figure 1: Schematic representation of a new document being added to the metadata storage. Full circles represent documents, arcs represent bibliographic references. Upon addition, the index is on one hand queried for bib. references (arcs) matching the added document (circle), and on the other hand it is queried for documents (circles) matching the bib. references of the added document (arcs).

of journal title, and year. For each of the hit, we check the remaining fields (possibly, as in the case of titles, using string distance functions). If we still cannot find matching entities, we weaken the query conditions to surnames and year alone, repeating the evaluation of hits.

3.3 Technical details

Bibliographic reference matching is implemented by two chains of processing nodes. In the first process, the source node `DateRangeItemRecordIteratorBuilder` iterates over all the item records in the repository. The next node, `ItemRecordToYElementConverterNode` accepts such item records on input and converts them to format accepted by indexing module. Next, `RelationsToElementsExpanderNode` takes bibliographic references in the input metadata records and promotes them to "first-class" records. This way all the bibliographic references can be indexed in the same way in which the main records are indexed in the system.

The second process also iterates over the item records in the repository. Next, using `ItemRecordToEnhancerMessageNode`, it wraps the records in messages that are used in further processing. Finally, the messages are piped to `BibReferenceMatchingWriterNode`, which matches the given document with other documents indexed in the system, and stores the results of the matching in the NLM format in `ext-link` element with attribute `ext-link-type` set to `eudml-item-id`.

3.4 Results

There are two areas of EuDML where bibliographic reference matching can be extremely useful. One is the user interface, where a user can click on a bibliographic reference

presented along with other metadata of a document, and thus navigate to the referenced document. The current EuDML web interface "understands" the results of bibliographic reference matches, and is able to present them. Another potential application of matched references is automated analysis of documents, where the network of bibliographic references reveals influential, often-cited documents. In this case, however, the results would not be directly visible to a user.

Since bibliographic reference matching is an internal process to enhance the metadata of articles and to setup the appropriate network structure it is embedded into the implementation within the Yadda framework.

However, for easing linking to EuDML items from external websites, it is very important that a simple yet efficient lookup interface is made available, as well as a machine-oriented batch for massive linking. Similar tools already exist as Crossref [2] services[1] for items endowed with a DOI. Based on preliminary bibliographic metadata aggregated in Task 3.4 and previous experience with CMD's mini-DML service, we have set up proof-of-concept demos of these services for EuDML.

- At `http://thar.ujf-grenoble.fr/cgi-bin/eulookup`, an interactive lookup where the user inputs a bibliographic citation (as a string) and gets back near matches when they are found (see Figures 2, 3).

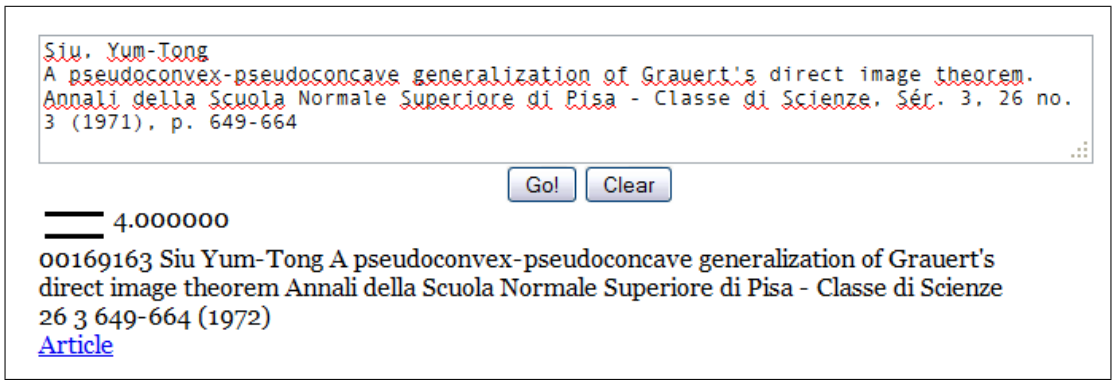

Figure 2: The EuDML interactive lookup: perfect match. Observe that the year in the provided reference is wrong.

- At `http://thar.ujf-grenoble.fr/cgi-bin/batcheulookup` a demo of a batch tool using the same matching engine. For this tool, the input format is an XML file conforming to EuDML schema, v. 1.0 as specified in deliverable D3.2. Each reference (**mixed-citation**) from the **ref-list** element is parsed and, when a match is found, an internal identifier is added to that reference and used to include an **ext-link** element providing the link to access the document from its provider.
- At `http://atlas.ujf-grenoble.fr/cgi-bin/zlookup` an example implementation of the reference matcher for the Zentralblatt Math database can be found.

1. See `http://www.crossref.org/guestquery/`.

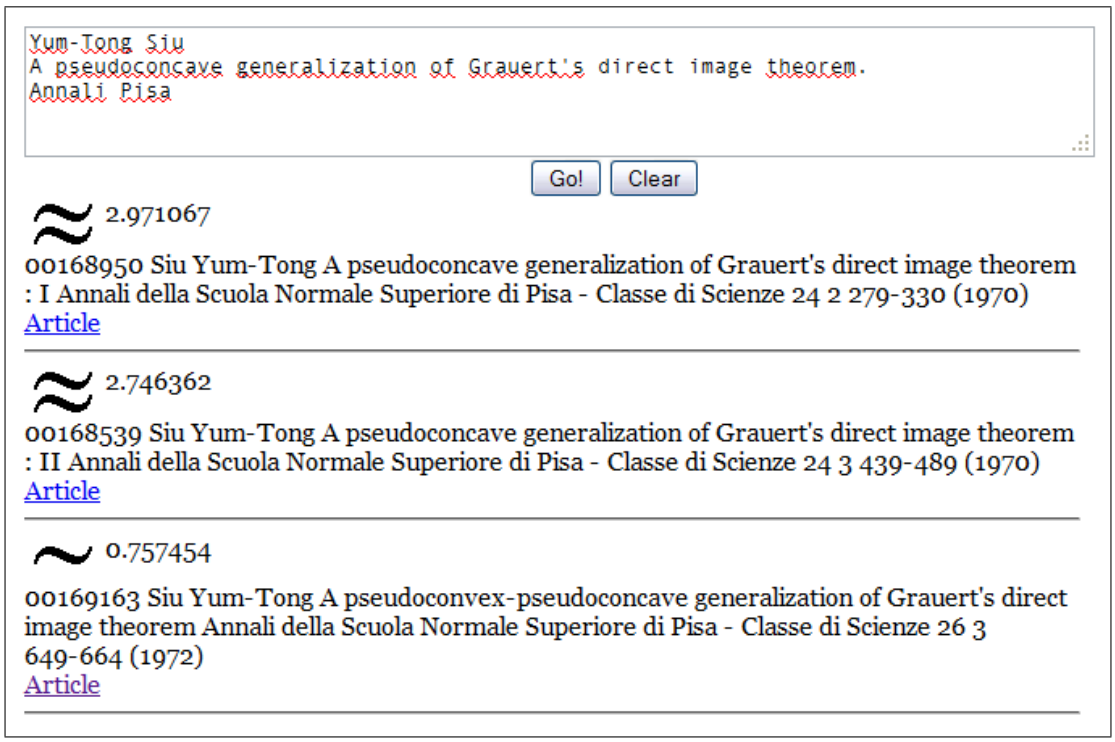

Figure 3: The EuDML interactive lookup: fuzzy match. These results would be difficult to exploit when linking is the goal, but the underdetermined reference should not be eligible for that purpose anyway. On the other hand, the returned list might be useful to a user looking for a work with a vague reference at hand.

It shows that the technology scales well (Zentralblatt hosting 20 times more items than EuDML), and that the accuracy benefits from the larger reference corpus, as a nearest match in a database registering almost all cited items is much less error prone.

The matching engine accepts a bibliographic reference as commonly found in the bibliography section of scientific articles. No semantic or typographic tagging of the reference's fields is needed (indeed, if it is present, it will be removed before further processing). The input is then analysed to extract relevant information such as names of authors, publication titles, etc. and composes these into an appropriate query for the central database. The returned results are in turn matched with the input query with respect to a particular distance metric with the aim of retrieving a small number of closely matching references only. After a successful query the results are returned together with the computed metric value given numerically (as a value between 0 and 4). In an interactive context, the closeness of the match may be indicated symbolically.

Observe that the lookup goes beyond what can be achieved by querying manually through a conventional search interface. Moreover, the distance metric ensures that the retrieved matches are indeed very close to the original only and that one therefore rarely gets more than one or two matching entries per query.

4 Evaluation and Further Work

Since a thorough evaluation of the services is planned in task 8.3 of workpackage 8 we will limit ourselves to only some basic observations in this section. They primarily serve to point out some of the further work that will need to be done on the integrated services.

Multilinguality Problems While the content provided in EuDML includes articles in a large number of languages there is not necessarily a large number of articles for every language. However, most of the techniques we integrate in this workpackage depend on a sufficiently large sample set to work with.

Consider, for instance, the gensim tool, which primarily works with demo data obtained from the arXiv. arXiv contains a majority of English articles but considerably less in other languages. For example, when looking for documents that are similar to an article in French one can easily obtain a number of very close matches since all French articles are attributed to a single topic and the only difference can stem from the respective English abstracts.

As multilinguality is a key feature in EuDML these issues will have to be addressed.

Metrics play an important role in the tools we provide. Therefore, thorough experimentation and evaluation of different metrics as well as combinations of metrics will have to be carried out. In the current implementation the used metrics are relatively strict and often far too syntax driven to allow for proper semantic association.

For example, when experimenting with bibliographic reference matching on the Zentralblatt database, one can easily observe some curious artifacts. Occasionally close matches for a particular article are ranked higher than the actual article. Similarly, the distance metric is particularly fragile with respect to changes in journal names. It gives higher rankings if the journal is given in the correct abbreviation while giving (sometimes significantly) rankings if the journal names are given in full or differently or only partially abbreviated.

This already indicates that both metrics and selected training data is currently too restrictive and has to be broadened in order to achieve semantically more appropriate results.

Query Corrections Currently the integrate tools will primarily be employed to produce enhanced metadata and pre-compute association networks in order to serve the users with articles related to a particular query or input document. However, in the light of EuDML's aim to improve accessibility one could imagine that similarity matching could also be exploited to provide query correction, e.g. for misspelled mathematical expressions, to support print impaired users. The similarity techniques could enable us to base these corrections on a mathematical corpus rather than using a standard dictionary and would require the introduction of some further degree of fuzziness into the matching algorithms. Clearly this part of the workpackage will then overlap with WP10.

References

[1] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] CrossRef. http://crossref.org/.
[3] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.
[4] gensim. http://nlp.fi.muni.cz/projekty/gensim/.
[5] Mark Lee, Petr Sojka, Volker Sorge, Josef Baker, Wojtek Hury, and Lukasz Bolikowski. Association Analyzer Implementation: State of the Art, November 2010. Deliverable D8.1 of EU CIP-ICT-PSP project 250503 EuDML: The European Digital Mathematics Library, http://eudml.eu/.
[6] Apache Lucene. http://lucene.apache.org/.