Enrichment and Structuring of Archival Description Metadata

Kalliopi Zervanou[†], Ioannis Korkontzelos[‡], Antal van den Bosch[†] and Sophia Ananiadou[‡]

[†] Tilburg centre for Cognition and Communication (TiCC), University of Tilburg
Warandelaan 2 - PO Box 90153, 5000 LE Tilburg, The Netherlands
{K.Zervanou, Antal.vdnBosch}@uvt.nl

[‡] National Centre for Text Mining, University of Manchester
131 Princess Street, Manchester M1 7DN, UK
{Ioannis.Korkontzelos, Sophia.Ananiadou}@manchester.ac.uk

Abstract

Cultural heritage institutions are making their digital content available and searchable online. Digital metadata descriptions play an important role in this endeavour. This metadata is mostly manually created and often lacks detailed annotation, consistency and, most importantly, explicit semantic content descriptors which would facilitate online browsing and exploration of available information. This paper proposes the enrichment of existing cultural heritage metadata with automatically generated semantic content descriptors. In particular, it is concerned with metadata encoding archival descriptions (EAD) and proposes to use automatic term recognition and term clustering techniques for knowledge acquisition and content-based document classification purposes.

1 Introduction

The advent of the digital age has long changed the processes and the media which cultural heritage institutions (such as libraries, archives and museums) apply for describing and cataloguing their objects: electronic cataloguing systems support classification and search, while cultural heritage objects are associated to digital metadata content descriptions. The expansion of the web and the increasing engagement of web users throughout the world has brought about the need for cultural heritage institutions to make their content available and accessible to a wider audience online.

In this endeavour, cultural heritage institutions face numerous challenges. In terms of metadata, different metadata standards currently exist for describing various types of objects, both within the same institution and across different institutions. Moreover, metadata object descriptions have been typically both created by and addressed to librarian and archivist experts who have been expected to assist visitors in their search. For this reason, they primarily refer to bibliographic descriptions (e.g. author/creator, title, etc.), or physical descriptions (e.g. size, shape, material, etc.), and location. The lack of semantic descriptors in this type of metadata makes it difficult for potential online visitors to browse and explore available information based on more intuitive content criteria.

Work on metadata in cultural heritage institutions has been largely focused on the issue of metadata heterogeneity. There have been efforts towards the development and adoption of collection-specific metadata standards, such as *MARC 21* (Library of Congress, 2010) and *EAD* (Library of Congress, 2002), for library and archival material respectively, which are intended to standardise metadata descriptions across different institutions. To address the issue of heterogeneity across different types of object collections, generic metadata schemas have been proposed, such as the *Dublin Core Metadata Initiative* (DCMI, 2011). Moreover, current research has attempted to integrate diverse metadata schemas by mappings across existing schemas (Bountouri and Gergatsoulis, 2009), or mappings of existing metadata to ontologies, either based on ad-hoc manually developed ontologies (Liao et al., 2010), or on existing standard ontologies for cultural heritage purposes (Lourdi et al., 2009), such as the *CIDOC Con-*

Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, pages 44–53,
Portland, OR, USA, 24 June 2011.

ceptual Reference Model (CIDOC, 2006). Other approaches attempt to address the issue of metadata heterogeneity from a pure information retrieval perspective and discard the diverse metadata structures in favour of the respective text content descriptions for full text indexing (Koolen et al., 2007). Zhang and Kamps (2009) attempt to exploit the existing metadata XML structure for XML-based retrieval, thus targeting individual document components. Similarly to our approach, they investigate metadata describing archive collections.

The work presented in this paper focuses on metadata for textual objects, such as archive documents, and on the issue of explicit, semantic, content descriptors in this metadata, rather than heterogeneity. In particular, we are concerned with the lack of explicit content descriptors which would support exploratory information search. For this purpose, we attempt to automatically enrich manually created metadata with content information. We view the problem from an unsupervised, text mining perspective, whereby multi-word terms recognised in free text are assumed to indicate content. In turn, the respective inter-relationships among the recognised terms in the hierarchy are assumed to reveal the knowledge structure of the document collection.

In this paper, we start with a description of our EAD dataset and the challenges which our dataset poses in text processing. Subsequently, we discuss our approach to the enrichment and structuring of these archival descriptions and present our experiments. We conclude with a discussion on our results and our considerations for future work.

2 EAD and Challenges in Text Processing

The Encoded Archival Description (EAD) was conceived as *"a nonproprietary encoding standard for machine-readable finding aids such as inventories, registers, indexes, and other documents created by archives, libraries, museums, and manuscript repositories to support the use of their holdings"* (Library of Congress, 2002). It is intended to be a data communication format based on SGML/XML syntax, aiming at supporting the accessibility to archival resources across different institutions and focusing on the structural content of the archival description, rather than its presentation. For this reason, the EAD schema is characterised by a hierarchical informational structure, where the deepest levels in the schema may inherit descriptive information defined in the upper levels. The schema defines a total of 146 elements. The three highest level elements are `<eadheader>`, `<frontmatter>`, and `<archdesc>`. `<eadheader>` is an element containing bibliographic and descriptive information about the metadata document, while `<frontmatter>` is an optional element describing the creation, publication, or use of the metadata document (Library of Congress, 2002). Both these two upper level elements do not contain information about the archival material itself. The designated element for this purpose is `<archdesc>` which describes *"the content, context, and extent of a body of archival materials, including administrative and supplemental information that facilitates use of the materials"* (Library of Congress, 2002).

EAD metadata files can be lengthy and complex in structure, with deep nesting of the XML hierarchy elements. As Zhang and Kamps (2009) also observe, the EAD elements may be of three types:

i. atomic units (or text content elements) which contain only text and no XML elements;
ii. composite units (or nested elements) which contain as nested other XML elements;
iii. mixed elements which contain both atomic and composite units.

The EAD documents used in this study describe archival collections of the International Institute of Social History (IISH). They are of varying length and are often characterised by long spans of non-annotated, free text. The degree of annotation, especially within *mixed* element types is inconsistent. For example, some names may be annotated in one element and others not, while quite often repeated mentions of the same name may not be annotated. Moreover, the text within an annotated element may include annotator comments (e.g., translations, alternate names, questions, notes, etc.), either in square brackets or parentheses, again in an inconsistent manner. The multilingual text content poses another challenge. In particular, the languages used in the description text vary, not only within a single EAD document, but often also within an element (mixed or atomic). In our approach, the former is addressed

by identifying the language at element level (cf. Section 3.2). However, the issue of mixed languages within an element is not addressed. This introduces errors, especially for multilingual elements of short text length.

3 Enrichment and Structuring Method

The overall rationale behind our method for the enrichment of EAD metadata with semantic content information is based on two hypotheses:

i. multi-word terms recognised in free text are valid indicators of content, and
ii. the respective term inter-relationships reflect the knowledge structure of the collection.

Thus, automatic term recognition and subsequent term clustering constitute the two core components of our EAD processing. In particular, as illustrated in Figure 1, we start with a pre-processing phase, where the EAD input SGML/XML files are first parsed, in order to retrieve the respective text content snippets, and then classified, based on language. Subsequently, terms are recognised automatically. The resulting terms are clustered as a hierarchy and, finally, the documents are classified according to the term hierarchy, based on the terms that they contain. To evaluate our term recognition process, we exploit knowledge from two sources: existing annotations in the EAD files, such as entity annotation residing in *mixed* elements (cf. Section 2) and entity and subject term information originating from the respective cultural heritage institution *Authority files*, namely the library files providing standard references for entities and terms that curators should use in their object descriptions. In this section, we discuss in more detail the methodology for each of the components of our approach.

3.1 EAD Text Element Extraction

In our processing of the EAD metadata XML, we focused on the free text content structured below the <archdesc> root element. As discussed in Section 2, it is the only top element which contains information about the archival material itself. In the *text element extraction* process, we parse the EAD XML and, from the hierarchically structured elements below <archdesc>, we select the text contained in <abstract>, <bioghist>, <scopecontent>, <odd>, <note> , <dsc> and <descgrp> and their nested elements.

Among these elements, the <dsc> (Description of Subordinate Components) provides information about the hierarchical groupings of the materials being described, whereas <descgrp> (DSC Group) defines nested encoded finding aids. They were selected because they may contain nested information of interest. The rest of the elements were selected because they contain important free text information related to the archive content:

- <bioghist>: describing the archive creator e.g. the life of the individual or family, or the administrative history of the organisation which created the archive;
- <scopecontent>: referring to the range and topical coverage of the described materials, often naming significant organisations, individuals, events, places, and subjects represented;
- <odd>: other descriptive data;
- <note>: referring to archivist comments and explanations;
- <abstract>: brief summaries of all the above information.

All other elements not referring to the archive semantic content, such as administrative information, storage arrangement, physical location, etc. were ignored. Moreover, atomic or composite elements without free text descriptions were not selected, because the descriptive information therein is assumed to be already fully structured.

3.2 Language Identification

As mentioned in Section 2, the languages used in the description text of the EAD documents vary, not only within a single EAD document, but often also within an EAD element. In our approach, the objective of the *language identification* process is to detect the language of the text content snippets, i.e. the output of the *text element extraction* process, and classify these snippets accordingly (cf. Figure 1).

Language identification is a text categorisation task, whereby identifiers attempt to learn the morphology of a language based on training text and, subsequently, use this information to classify unknown text accordingly. For this reason, training a language identification component requires a training corpus for each language of interest.

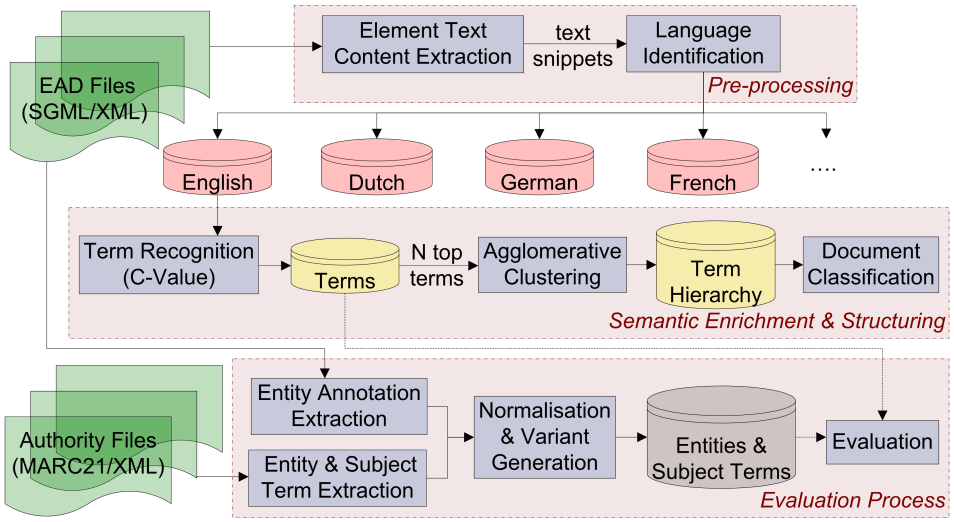


Figure 1: Block diagram of EAD metadata enrichment and structuring process

Computational approaches to language identification can be coarsely classified into information-theoretic, word-based, and N-gram-based. Information-theoretic approaches compare the compressibility of the input text to the compressibility of text in the known languages. Measuring compressibility employs mutual information measures (Poutsma, 2002). Word-based approaches consider the amount of common words or special characters between the input text and a known language. Finally, N-gram-based approaches construct language models beyond word boundaries, based on the occurrence statistics of N-grams up to some predefined length N (Dunning, 1994). The subsequent language identification in unknown text is based on the similarity of the unknown text N-gram model to each training language model.

As evidenced by these approaches, language identification relies on some form of comparison of the unknown text to known languages. For this reason, the respective text categorisation into a given language suffers when the input text is not long enough: the shorter the input text is, the fewer the available features for comparison against known language models. Moreover, errors in the categorisation process are also introduced, when the language models under comparison share the same word forms.

In our approach, we have opted for the most popular language identification method, the one based on N-grams. Nevertheless, any other language identification method could have been applied.

3.3 Term Recognition

The objective of *term recognition* is the identification of linguistic expressions denoting specialised concepts, namely domain or scientific terms. For information management and retrieval purposes, the automatic identification of terms is of particular importance because these specialised concept expressions reflect the respective document content.

Term recognition approaches largely rely on the identification of term formation patterns. Linguistic approaches use either syntactic (Justeson and Katz, 1995; Hearst, 1998), or morphological (Heid, 1998) rule patterns, often in combination with terminological or other lexical resources (Gaizauskas et al., 2000) and are typically language and domain specific.

Statistical approaches typically combine linguistic information with statistical measures. These measures can be coarsely classified into two categories: *unithood-based* and *termhood-based*. *Unithood-based* approaches measure the attachment strength among the constituents of a candidate term. For example, some unithood-based measures are frequency of co-occurrence, hypothesis testing statistics, log-likelihood ratios test (Dunning, 1993) and pointwise mutual information (Church and Hanks, 1990). *Termhood-based* approaches attempt to measure the degree up to which a candidate expression is a valid term, i.e. refers to a specialised concept. They attempt to measure this degree by considering *nestedness* information, namely the fre-

quencies of candidate terms and their subsequences. Examples of such approaches are C-Value and NC-Value (Frantzi et al., 2000) and the statistical barrier method (Nakagawa, 2000).

It has been experimentally shown that *termhood-based* approaches to automatic term extraction outperform *unithood-based* ones and that *C-Value* (Frantzi et al., 2000) is among the best performing *termhood-based* approaches (Korkontzelos et al., 2008). For this reason, we choose to employ the *C-Value* measure in our pipeline. *C-Value* exploits nestedness and comes together with a computationally efficient algorithm, which scores candidate multi-word terms according to the measure, considering:

- the total frequency of occurrence of the candidate term;
- the frequency of the candidate term as part of longer candidate terms;
- the number of these **distinct** longer candidates;
- the length of the candidate term (in tokens).

These arguments are expressed in the following nestedness formula:

$$N(\alpha) = \begin{cases} f(\alpha), \text{ if } \alpha \text{ is not nested} \\ f(\alpha) - \frac{1}{|T_\alpha|} \sum_{b \in T_\alpha} f(b), \text{ otherwise} \end{cases} \quad (1)$$

where α is the candidate term, $f(\alpha)$ is its frequency, T_α is the set of candidate terms that contain α and $|T_\alpha|$ is the cardinality of T_α. In simple terms, the more frequently a candidate term appears as a substring of other candidates, the less likely it is to be a valid term. However, the greater the number of **distinct** term candidates in which the target term candidate occurs as nested, the more likely it is to be a valid term. The final *C-Value* score considers the length ($|\alpha|$) of each candidate term (α) as well:

$$C\text{-}value(\alpha) = \log_2 |\alpha| \times N(\alpha) \quad (2)$$

The C-Value method requires linguistic pre-processing in order to detect syntactic term formation patterns. In our approach, we used LexTagger (Vasilakopoulos, 2003), which combines transformation-based learning with decision trees and we adapted its respective lexicon to our domain. We also included WordNet lemma information in our processing, for text normalisation purposes. Linguistic pre-processing is followed by the computation of C-Value on the candidate terms, in length order, longest first. Candidates that satisfy a C-Value threshold are sorted in decreasing C-Value order.

3.4 Hierarchical Agglomerative Clustering

In our approach, term recognition provides content indicators. In order to make explicit the knowledge structure of the EAD, our method requires some form of concept classification and structuring. The process of *hierarchical agglomerative clustering* serves this objective.

Agglomerative algorithms are very popular in the field of *unsupervised concept hierarchy induction* and are typically employed to produce unlabelled taxonomies (King, 1967; Sneath and Sokal, 1973). *Hierarchical clustering* algorithms are based on measuring the distance (dissimilarity) between pairs of objects. Given an object distance metric D, the similarity of two clusters, $\mathcal{A}$ and $\mathcal{B}$, can be defined as a function of the distance D between the objects that the clusters contain. According to this similarity, also called *linkage criterion*, the choice of which clusters to merge or split is made. In our approach, we have experimented with the three most popular criteria, namely:

Complete linkage (CL): The similarity of two clusters is the maximum distance between their elements

$$sim_{CL}(\mathcal{A}, \mathcal{B}) = \max_{x \in \mathcal{A}, y \in \mathcal{B}} D(x, y) \quad (3)$$

Single linkage (SL): The similarity of two clusters is the minimum distance between their elements

$$sim_{SL}(\mathcal{A}, \mathcal{B}) = \min_{x \in \mathcal{A}, y \in \mathcal{B}} D(x, y) \quad (4)$$

Average linkage (AL): The similarity of two clusters is the average distance between their elements

$$sim_{AL}(\mathcal{A}, \mathcal{B}) = \frac{1}{|\mathcal{A}| \times |\mathcal{B}|} \sum_{x \in \mathcal{A}} \sum_{y \in \mathcal{B}} D(x, y) \quad (5)$$

To estimate the distance metric D we use either the *document co-occurrence* or the *lexical similarity* metric. The chosen distance metric D and linkage criterion are employed to derive a hierarchy of terms by agglomerative clustering.

Our *document co-occurrence (DC)* metric is defined as the number of documents (d) in the collection (R) in which both terms (t_1 and t_2) co-occur:

$$DC = \frac{1}{|R|} |\{d : (d \in R) \wedge (t_1 \in d) \wedge (t_2 \in d)\}| \quad (6)$$

The above metric accepts that the distance between two terms is inversely proportional to the number of documents in which they co-occur.

Lexical Similarity (LS), as defined in Nenadić and Ananiadou (2006), is based on shared term constituents:

$$LS = \frac{|P(h_1) \cap P(h_2)|}{|P(h_1)| + |P(h_2)|} + \frac{|P(t_1) \cap P(t_2)|}{|P(t_1)| + |P(t_2)|} \quad (7)$$

where t_1 and t_2 are two terms, h_1 and h_2 their heads, $P(h_1)$ and $P(h_2)$ their set of head words, and $P(t_1)$ and $P(t_2)$ their set of constituent words, respectively.

3.5 Document Classification

The term hierarchy is used in our approach for semantic classification of documents. In this process, we start by assigning to each leaf node of the term hierarchy the set of EAD documents in which the corresponding term occurs. Higher level nodes are assigned the union of the document sets of their daughters. The process is bottom-up and applied iteratively, until all hierarchy nodes are assigned a set of documents.

Document classification, i.e. the assignment of document sets to term hierarchy nodes, is useful, among others, for structured search and indexing purposes. Moreover, it provides a direct soft-clustering of documents based on semantics, given the number of desired clusters, C. C corresponds to a certain horizontal cut of the term hierarchy, so that C top nodes appear, instead of one. The document sets assigned to these C top nodes represent the C desired clusters. This document clustering approach is *soft*, since each document can occur in one or more clusters.

3.6 Evaluation Process

The automatic *evaluation process*, illustrated in Figure 1, serves the purpose of evaluating the *term recognition* accuracy. Since the objective of term recognition tools is the detection of linguistic expressions denoting specialised concepts, i.e. terms, the results evaluation would ideally require input from the respective domain experts. This is a laborious and time consuming process which also entails finding the experts willing to dedicate effort and time for this task. In response to this issue, we decided to exploit the available domain-specific knowledge resources and automate part of the evaluation process by comparing our results to this existing information. Thus, the automatic *evaluation process* is intended to give us an initial estimate of our performance and reduce the amount of results requiring manual evaluation. The available resources used are of two types:

i. entity annotations in the EAD documents (i.e. names of persons, organisations and geographical locations);
ii. entity and subject terms originating from the cultural heritage institution *Authority files*.

The entity annotations in the EAD documents were not considered during our *term recognition*. The entity and subject terms of the respective *Authority file* records are encoded in MARC21/XML format (Library of Congress, 2010). MARC (MAchine-Readable Cataloging) is a standard initiated by the US Library of Congress and concerns the representation of bibliographic information and related data elements used in library catalogues. The MARC21 Authority files resource used in our evaluation provides, among other information, the standard references for entities and the respective possible entity reference variations, such as alternate names or acronyms, etc., that curators should use in their object descriptions. The subject term Authority records provide mappings between a legacy subject term thesaurus which is no longer used for classification, and current library records.

In the *evaluation process* the EAD SGML/XML and the MARC21/XML Authority files are first parsed by the respective parsers in order to extract the XML elements of interest. Subsequently, the text-content of the elements is processed for normalisation and variant generation purposes. In this process, normalisation involves cleaning up the text from intercepted comments and various types of inconsistent notes, such as dates, aliases and alternate names, translations, clarifications, assumptions, questions, lists, etc. Variant generation involves detecting the acronyms, abbreviated names and aliases mentioned in the element text and creating the reversed variants for, e.g., `[Last_Name, First_Name]` sequences. The results of this process, from both EAD and Authority files, are merged into a single list for every respective category (or-

language	snippets	language	snippets
Dutch	50,363	Spanish	3,430
German	41,334	Danish	2,478
English	19,767	Italian	1,100
French	6,182	Swedish	699

Table 1: Number of snippets per identified language.

ganisations, persons, geographic locations and subject terms) and are compared to our term results list.

4 Experimental Setting

For training the *language identification* component, we used the European Parliament Proceedings Parallel Corpus (Europarl) which covers the proceedings of the European Parliament from 1996 to 2006 (Koehn, 2005). The corpus size is 40 million words per language and is translated in Danish, German, Greek, English, Spanish, Finnish, French, Italian, Dutch, Portuguese and Swedish. In our experiments, we take as input for subsequent term recognition only the snippets identified as English text.

In the experiments reported in this work, we accept as term candidates morpho-syntactic pattern sequences which consist of adjectives and nouns, and end with a noun. The *C-Value* algorithm (cf. Section 3.3) was implemented under two different settings:

i. one only considering as term candidates adjective and noun sequences that appear at least once as non-nested in other candidate terms; and
ii. one that considers all adjective and noun sequences, even if they never occur as non-nested.

Considering that part-of-speech taggers usually suffer high error rates when applied on specialty domains, the former setting is expected to increase precision, whereas the latter to increase recall (cf. Section 5).

We accepted as valid terms all term candidates whose *C-Value* score exceeds a threshold, which was set to 3.0 after experimentation. In the subsequent hierarchical agglomerative clustering process, we experimented with all six combinations of the three linkage criteria (i.e. *complete*, *single* and *average*) with the two distance metrics (i.e. *document co-occurrence* and *lexical similarity*) described in Section 3.4.

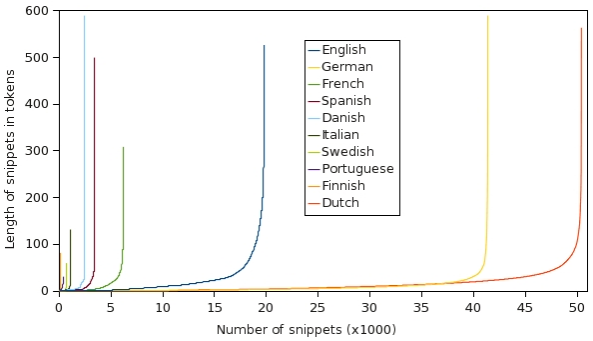


Figure 2: Length of snippets per identified language.

5 Results

The EAD document collection used for this study consisted of $3,093$ SGML/XML files. As shown on Table 1, according to our language identifier, the majority of the text snippets of the selected EAD XML elements were in Dutch, followed by German and English. We selected for later processing $19,767$ snippets classified as English text, corresponding to $419,857$ tokens. A quantitative evaluation of the language identifier results has not been performed. However, our observation of the term recognition results showed that there were some phrases, mostly Dutch and German entity names (organisations and persons mostly) classified as English. This might be due to these entities appearing in their original language within English text, as it is often the case in our EAD files. Moreover, manual inspection of our results showed that other languages classified as English, e.g. Turkish and Czech, were not covered by Europarl.

As mentioned in Section 3.2, short text snippets may affect language identification performance. Figure 2 illustrates the snippet length per identified language. We observe that the majority of text snippets is below 10 tokens, few fall within an average length of 20 to 50 tokens approximately, and very few are above 100 tokens.

Figure 3 shows the results of our automatic evaluation for the term recognition process. In this graph, the upper, red curve shows the percentage of correct terms for the C-Value setting considering as term candidates adjective and noun sequences that appear at least once as non-nested in other candidate terms. The lower, blue curve shows the per-

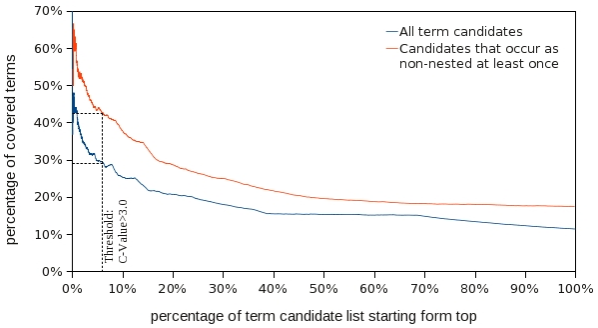


Figure 3: Term coverage for each C-Value setting based on EAD & Authority entity and subject term evaluation.

centage of correct terms for the C-Value setting considering all adjective and noun sequences, even if they never occur as non-nested. In this automatic evaluation, *correct* terms are, as presented in Section 3.6, those candidate terms matching the combined lists of entity and subject terms acquired by the respective EAD and MARC21 Authority files. We observe that the C-Value setting which considers only noun phrase patterns occurring at least once as non-nested, displays precision up to approximately 70% for the top terms in the ranked list, whereas the other setting considering all noun phrase sequences, reaches a maximum of 49%. The entire result set above the 3.0 C-Value threshold amounts to $1,345$ and $2,297$ terms for each setting, and reaches precision of 42.01% and 28.91% respectively. Thus, regarding precision, the selective setting clearly outperforms the one considering all noun phrases, but it also reaches a lower recall, as indicated by the actual terms within the threshold. We also observe that precision drops gradually below the threshold, an indication that the ranking of the C-Value measure is effective in promoting valid terms towards the top. This automatic evaluation considers as erroneous unknown terms which may be valid. Further manual evaluation by domain experts is required for a more complete picture of the results.

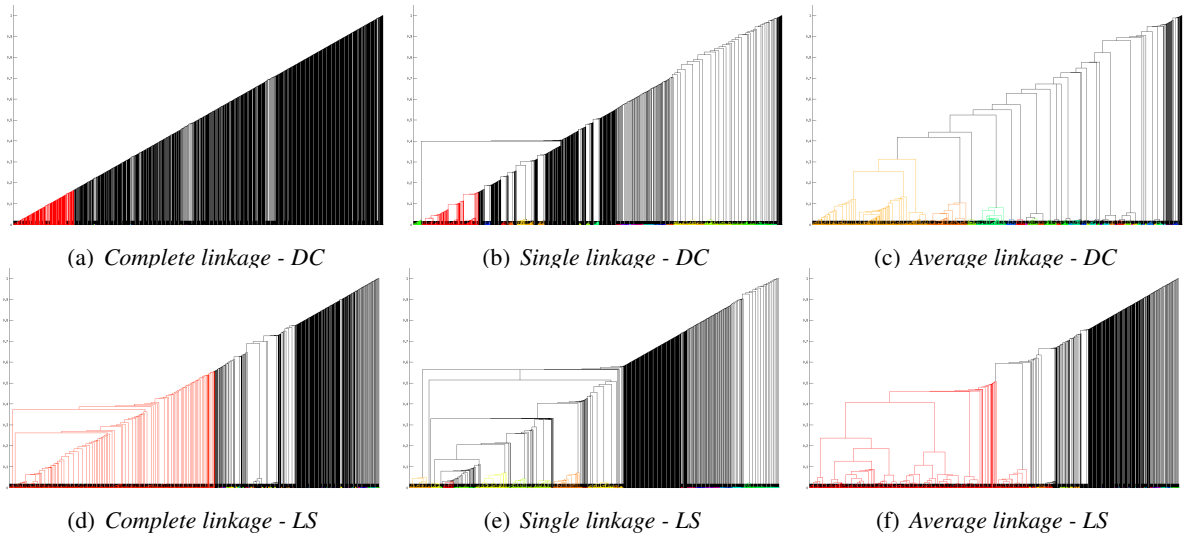
Figure 4 shows six dendrograms, each representing the term hierarchy produced by the respective combination of linkage criterion to distance metric. The input for these experiments consists of all terms exceeding the C-Value threshold, and by considering only noun phrase sequences appearing at least once as non-nested. Since the hierarchies contain $1,345$ terms, the dendrograms are very dense and difficult to inspect thoroughly. However, we include them based on the fact that the overall shape of the dendrogram can indicate how much narrow or broad the corresponding hierarchy is and indirectly its quality. Narrow here characterises hierarchies whose most non-terminal nodes are parents of one terminal and one non-terminal node. Narrow hierarchies are deep while broader hierarchies are shallower.

Broad and shallow hierarchies are, in our case, of higher quality, since terms are expected to be related to each other and form distinct groups. In this view, average linkage leads to richer hierarchies (Figures 4(c), 4(f)), followed by single linkage (Figures 4(b), 4(e)) and, finally, complete linkage (Figures 4(a), 4(d)). The hierarchy of higher quality seems to be the result of average linkage and *document co-occurrence* combination (Figure 4(c)), followed by the combination of average linkage and *lexical similarity* (Figure 4(f)). Clearly, these two hierarchies need to be investigated manually and closely to extract further conclusions. Moreover, an application-based evaluation could investigate whether different clustering settings suit different tasks.

6 Conclusion and Future Work

In this paper, we have presented a methodology for semantically enriching archival description metadata and structuring the metadata collection. We consider that terms are indicators of content semantics. In our approach, we perform term recognition and then hierarchically structure the recognised terms. Finally, we use the term hierarchy to classify the metadata documents. We also propose an automatic evaluation of the recognised terms, by comparing them to domain knowledge resources.

For term recognition, we used the C-Value algorithm and found that considering noun phrases which appear at least once independently, outperforms considering *all* noun phrases. Regarding hierarchical clustering, we observe that the average linkage criterion combined with a distance metric based on document co-occurrence produces a rich broad hierarchy. A more thorough evaluation of these results is required. This should include a manual evaluation of recognised terms by domain experts and an application-based evaluation of the resulting document classification.



(a) *Complete linkage - DC* (b) *Single linkage - DC* (c) *Average linkage - DC*

(d) *Complete linkage - LS* (e) *Single linkage - LS* (f) *Average linkage - LS*

Figure 4: Dendrograms showing the results of agglomerative clustering for all linkage criteria and distance metrics, *document co-occurrence (DC)* and *Lexical Similarity (LS)*.

References

Lina Bountouri and Manolis Gergatsoulis. 2009. Interoperability between archival and bibliographic metadata: An EAD to MODS crosswalk. *Journal of Library Metadata*, 9(1-2):98–133.

Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

CIDOC. 2006. The CIDOC Conceptual Reference Model. CIDOC Documentation Standards Working Group, International Documentation Committee, International Council of Museums. `http://www.cidoc-crm.org/`.

DCMI. 2011. The Dublin Core Metadata Initiative. `http://dublincore.org/`.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Ted Dunning. 1994. Statistical identification of language. MCCS 94-273. Technical report, Computing Research Laboratory, New Mexico State University.

Katerina Frantzi, Sophia Ananiadou, and Hideki Mima. 2000. Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries*, 3(2):115–130.

Robert Gaizauskas, George Demetriou, and Kevin Humphreys. 2000. Term recognition in biological science journal articles. In *Proc. of the NLP 2000 Workshop on Computational Terminology for Medical and Biological Applications*, pages 37–44, Patras, Greece.

Marti Hearst. 1998. Automated discovery of WordNet relations. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 131–153. MIT Press.

Ulrich Heid. 1998. A linguistic bootstrapping approach to the extraction of term candidates from german text. *Terminology*, 5(2):161–181.

John Justeson and Slava Katz. 1995. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(1):9–27.

Benjamin King. 1967. Step-Wise clustering procedures. *Journal of the American Statistical Association*, 62(317):86–101.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Machine Translation Summit X*, pages 79–86, Phuket, Thailand.

Marijn Koolen, Avi Arampatzis, Jaap Kamps, Vincent de Keijzer, and Nir Nussbaum. 2007. Unified access to heterogeneous data in cultural heritage. In *Proc. of RIAO '07*, pages 108–122, Pittsburgh, PA, USA.

Ioannis Korkontzelos, Ioannis Klapaftis, and Suresh Manandhar. 2008. Reviewing and evaluating automatic term recognition techniques. In Bengt Nordström and Aarne Ranta, editors, *Proc. of GoTAL '08*, volume 5221 of *LNCS*, pages 248–259, Gothenburg, Sweden. Springer.

Shu-Hsien Liao, Hong-Chu Huang, and Ya-Ning Chen. 2010. A semantic web approach to heterogeneous metadata integration. In Jeng-Shyang Pan, Shyi-Ming Chen, and Ngoc Thanh Nguyen, editors, *Proc. of ICCCI '10*, volume 6421 of *LNCS*, pages 205–214, Kaohsiung, Taiwan. Springer.

Library of Congress. 2002. Encoded archival description (EAD), version 2002. Encoded Archival Description Working Group: Society of American Archivists,

Network Development and MARC Standards Office, Library of Congress. http://www.loc.gov/ead/.

Library of Congress. 2010. MARC standards. Network Development and MARC Standards Office, Library of Congress, USA. http://www.loc.gov/marc/index.html.

Irene Lourdi, Christos Papatheodorou, and Martin Doerr. 2009. Semantic integration of collection description: Combining CIDOC/CRM and Dublin Core collections application profile. *D-Lib Magazine*, 15(7/8).

Hiroshi Nakagawa. 2000. Automatic term recognition based on statistics of compound nouns. *Terminology*, 6(2):195–210.

Goran Nenadić and Sophia Ananiadou. 2006. Mining semantically related terms from biomedical literature. *ACM Transactions on Asian Language Information Processing (TALIP)*, 5(1):22–43.

Arjen Poutsma. 2002. Applying monte carlo techniques to language identification. *Language and Computers*, 45:179–189.

Peter Sneath and Robert Sokal. 1973. *Numerical taxonomy: the principles and practice of numerical classification*. Freeman, San Francisco, USA.

Argyris Vasilakopoulos. 2003. Improved unknown word guessing by decision tree induction for POS tagging with tbl. In *Proc. of CLUK '03*, Edinburgh, UK.

Junte Zhang and Jaap Kamps. 2009. Focused search in digital archives. In Gottfried Vossen, Darrell D. E. Long, and Jeffrey Xu Yu, editors, *Proc. of WISE '09*, volume 5802 of *LNCS*, pages 463–471, Poznan, Poland. Springer.