Evaluating Tools for Automatic Concept Extraction: a Case Study from the Musicology Domain

[1]Scott Piao, [2]Jamie Forth, [1]Ricardo Gacitua, [1]Jon Whittle and [2]Geraint Wiggins
[1]Computing Department
Lancaster University
Lancaster, UK
{s.piao,r.gacitua,j.n.whittle@lancaster.ac.uk}

[2]Department of Computing &
Centre for Cognition, Computation and Culture
Goldsmiths, University of London, London, UK
{j.forth,g.wiggins@gold.ac.uk}

ABSTRACT
Term extraction algorithms have various applications in Digital Economy research with the rise of online sources. This paper reports on an evaluation of five term extraction algorithms for automatic concept extraction in the musicology domain, which is carried out in the context of the RCUK funded SerenA Project. Our focus here is to identify the algorithms that are most suitable for the task of concept extraction. In our evaluation, the *C*-value algorithm produced the best result, while others achieved encouraging performances revealing interesting features of each algorithm that will be helpful for developing better algorithms.

General Terms
Algorithms, Performance, Experimentation, Languages.

Keywords
Term Extraction, Concept Extraction, Serendipitous Connection

1. Introduction
In this paper we report on an evaluation of a set of tools for automatic concept extraction in the musicology domain. This work is carried out in the context of the UKRC Digital Economy funded SerenA Project, which is investigating new algorithms for proactively suggesting serendipitous connections between researchers from different disciplines, based on a conceptual analysis of the researchers' publications, blogs and online notes. An important task in the project is to automatically extract key concepts from textual documents (e.g., research publications) in a given domain. Later, SerenA will make non-obvious connections between researchers in academic disciplines by semantically matching concepts from different domains. For example, some researchers of financial studies can be connected to linguists as their works may share some concepts of Natural Language Processing. Our focus here is to identify the algorithms that are most suitable for the task of concept extraction. We use the musicology domain as a test study because it will form a major domain of study in SerenA. While there are numerous reports on evaluation of term extraction tools, very few evaluation studies have been conducted on the musicology domain. For our evaluation, we selected five term extraction tools, which employ some of the most efficient algorithms reported. The tools were tested on a collection of musicology publications containing approximately 315,000 words. In our evaluation, the C-value algorithm produced the best results while others achieved encouraging performance. Our evaluation reveals some interesting features of the algorithms which will be helpful for improving and developing better algorithms.

Digital Futures'10, October 11 - 12, 2010, Nottingham, UK.

2. Term Extraction Algorithms for Evaluation
We selected five term extraction tools employing different algorithms which are well documented in publications. These five algorithms are capable of extracting both single and multi-word terms using single metrics. Below are brief descriptions of these algorithms.

1) Weirdness: Ahmad et al. [1] proposed a *weirdness* indexing algorithm for a document retrieval system. They suggested that domain topics can be identified by quantitatively comparing differences between general language and special language texts, because the word occurrence probability of the domain-related lexical items would tend to be higher in the specialist text than in the general text. They used BNC [6] as the general language text, or reference corpus. Given a candidate term, the *weirdness* score is calculated by dividing the probability of a term in the specialist text by that in the reference corpus.

2) Glossary Extraction: Kozakov et al. [5] reported an application of extracting domain specific glossaries from document collections used as a component of the IBM Textract system. They mainly consider noun phrases and non-auxiliary verbs, including both single word and multi-word units. NLP tools such as a morphological analyser and a Part-of-speech (POS) tagger as well as a POS pattern filter are used to extract candidate terms. Two measures, domain specificity and term cohesion, are used to jointly determine the "goodness" of candidate terms. The relative probability of words in the domain specific and general corpora is used to estimate the term's domain specificity while a generalized Dice Coefficient [8] is used to measure the term cohesion. These two measures are weighted and combined together to determine the "goodness" of candidate terms.

3) Term Extraction: Sclano and Velardi [7] designed another algorithm to extract domain terms. It consists of a linguistic processor and a set of filters. Given an input text, the linguistic processor is used to produce candidate terms by selecting typical terminological structures, such as compounds, adjective-nouns etc., which are pruned using three main statistical filters: a) domain pertinence filter, b) domain consensus filter, and c) lexical cohesion filter. Some heuristic information is also used to enhance the filters, such as structural relevance, misspelling, etc. The three main filter scores are weighted and combined to yield the final filtering metric.

4) C-Value: Frantzi et al. [2] presented the C-value method for extracting multiword terms. Their algorithm first uses a POS tagger and a POS pattern filter to collect noun phrases as candidate terms. Next, the statistical measure C-value is used to determine the termhood and unithood of the candidates.
(*See Formula 1: C-value*)

where $f(.)$ denotes frequency of items, T_α the set of extracted terms containing α and $P(T_\alpha)$, the number of these candidate terms. C-value has become a popular measure for automatic term extraction, and some modified versions are capable of extracting both single word and multi-word terms.

5) RAI: Gacitua et al. [3] implemented an algorithm named relevance-driven abstraction identification (RAI) as a component of the MaTREx system [4]. This algorithm proceeds as follows: a) The input text is POS tagged, filtered out stop words and lemmatised (convert inflectional word variants into base forms); b) Each word is assigned a log-likelihood score by applying corpus-based frequency profiling; c) POS pattern filters are used to identify candidate multi-word terms; d) A significance score is calculated for each multi-word term by combining the log-likelihood scores of its constituent words; e) The candidate terms, single word and multiword terms combined, are ranked with the significance scores.

In our experiment, we used a package implementing the first four algorithms that was developed by Zhang et al. [9] in their comparative evaluation work. On the other hand, the RAI tool was developed by Gacitua et al. [3]. For the NLP processing, the OpenNLP package (http://sourceforge.net/projects/opennlp/) is used. A common feature of the above tools except C-value is that they all use a reference corpus for measuring termhood: BNC corpus in this case.

3. Evaluation

3.1 Test data

As mentioned earlier, one of our tasks is to extract domain knowledge from individual people's moderate sized document collections. Therefore, we limited the size of the test data to 27 systematic musicology papers, containing about 315,000 words. In order to guarantee the quality of the data, a musicologist (one of the authors) selected representative publications from within his specific domain of musicological expertise. The original documents are pdf files, so we extracted plain text from them using MultiValent (http://multivalent.sourceforge.net/). The automatically converted text contained some noise, such as broken words, broken tables etc., which caused some extra errors in the term extraction. This problem will be addressed by improving the conversion tools and developing text clearing tools for the document collection stage of the work.

3.2 Discussion of tool performance

In our experiment, the test data was processed using the five tools and the top 100 items from the resultant term lists were manually examined by the domain expert. In details, the results were examined twice using different criteria. In the first round, the names of institutions, organizations, publications and authors were counted as domain related terms; in the second round they were counted as errors. Whether the names are part of domain-related terms or not is an issue for further discussion, but they obviously provide important information pertinent to the contents. Therefore, the two sets of evaluation results reflect the useful capability of the tools for extracting different types of terms. Table 1 shows the evaluation statistics, where the names *wilder*, *glossex*, *termex*, *c-value* and *rai* respectively correspond to the algorithms from 1) to 5) described in section 2, and check1 and check2 correspond to the two rounds of the checking.

(*See Table 1: Precisions of the term extraction tools*)

As shown in the table, the C-value algorithm produced the best and most stable results for both of the checking criteria, and the RAI algorithm produced the same precision for both criteria, as it did not extract any names in the top 100 items. The other three tools show fluctuations of performance when different criteria are applied. For example, 48 and 47 among the top 100 terms extracted by *wilder* and *glossex* are names, indicating that they can be effective tools for named entity identification. Another surprising finding is that *termex*, one of the best performing algorithms in Zhang et al.'s evaluation [9], performed rather poorly in our experiment. This suggests that algorithms may exhibit different performance within different domains.

4. Conclusion

We evaluated five tools/algorithms for automatic musicology concept extraction, focusing on the examination of the performance of the various algorithms on this domain. In our study, the C-value algorithm demonstrated the most stable performance with the highest precision. Note that, among the five tools, C-value is the only one without a reference corpus. There is an interesting issue of how the reference corpus approach would affect the performance, if we combine the relative probability measure with the C-value algorithm. Term extraction algorithms have wide-ranging applications in Digital Economy research – with the rise of social networks, blogs and other online sources, there is a wide body of text available for analysis that is useful for a range of purposes.

References

[1] Ahmad, K., Gillam, L., and Tostevin, L. 1999. University of surrey participation in trec8: Weirdness indexing for logical document extrapolation and retrieval (wilder). In *Proceedings of the TREC-8* (Gaithersburg, Maryland).

[2] Frantzi, K.T., Ananiadou, S., and Mima, H. 2000. Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries*. 3, 2, 115–130.

[3] Gacitua, R., Sawyer, P., and Gervasi, Vincenzo. 2010. On the effectiveness of abstraction identification in requirements engineering. In *Proceedings of the 18th International IEEE Requirements Engineering Conference (RE'10)* (Sydney, Australia). (In press).

[4] Gacitua, R, , Ma, L., Nuseibeh, B., Piwek, P., de Roeck, A.N., Rouncefield, M., Sawyer,P., Willis, A and Yang, H. 2009. Making tacit requirements explicit. Second International Workshop on MaRK'09, Atlanta, USA.

[5] Kozakov, L., Park, Y., Fin, T., Drissi, Y., Doganata, Y., and Cofino, T. 2004. Glossary extraction and utilization in the information search and delivery system for IBM technical support. *IBM Systems Journal*. 43, 3, 546–563.

[6] Leech, G. 1992. 100 million words of English: the British National Corpus. *Language Research.* 28, 1, 1-13.

[7] Sclano, F., and Velardi, P. 2007. TermExtractor: a web application to learn the shared terminology of emergent web communities. In *Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (I-ESA 2007)* (Funchal, Madeira Island, Portugal).

[8] van. Rijsbergen, C. J. 1979. Information Retrieval. London: Butterworths.

[9] Zhang, Z., Iria, J., Brewster, C., and Ciravegna, F. 2008. A comparative evaluation of term recognition algorithms. In *Proceedings of the LREC 2008* (Marrakech, Morocco).

Formula 1: *C*-value

$$\text{c-value} = \begin{cases} \log_2 |\alpha| \cdot f(\alpha) & \text{if } \alpha \text{ is not nested,} \\ \log_2 |\alpha| \cdot \left[f(\alpha) - \frac{1}{P(T_\alpha)} \sum_{b \in T_\alpha} f(b) \right] & \text{otherwise} \end{cases}$$

Table 1: Precisions of the term extraction tools

eval.	**wilder**	**glossex**	**termex**	**c-value**	**rai**
check1	89%	**94%**	76%	**94%**	87%
check2	41%	47%	62%	88%	87%