Workshop on Data Analysis WDA'2008

František Babič, Ján Paralič, Andreas Rauber (Eds.)

Proceedings of the
8th International Student Workshop

Dedinky, Slovakia

June 26 – 29, 2008

Preface

The 8th International Student Workshop on Data Analysis (WDA) took place in Dedinky, a beautiful location on the south border of Slovak Paradise. After one year break the Workshop has moved from its traditional structure of participants, where majority of graduate students have been active, to a new model of contributions where a majority of them have been delivered by PhD-students, this year from two different universities - Technical University of Košice and Vienna University of Technology.

The Workshop, held on June 26 - 29, 2008, consisted of three main thematic sessions. The first session was focused on text analysis in general and various aspects of text classification in particular. The second session brought the new topic of digital preservation into the WDA palette. Finally, the last session was more technically oriented on various approaches to realisation of midleware supporting text analysis in the semantic web architecture.

The first session about text mining started with a paper presented by a fresh master in artificial intelligence, Gabriel Tutoky, who presented a meta-learning approach for textual document classification task and an automatic selection of the best available algorithm for creation of classifiers. The next contribution by PhD student Peter Smatana focussed on approaches for increasing document classification accuracy by dividing documents into smaller parts and taking into account the distribution of terms within the document itself. the last presentation in this section provided by Pavol Jasem described preliminary thoughts about a new project in biomedical application area, where information retrieval and text mining approache are going to be exploited.

The next session switched to a new and currently very active area of research and practice, namely digital preservation. In this section, PhD student Christoph Becker presented extensible integration architecture for automating the analysis and evaluation of potential preservation actions. In the second presentation PhD student Mark Guttenbrunner identified significant properties of different types of interactive objects in order to determine optimal preservation solutions. The last presentation was given by Jakob Frank, who presented a client-server system for access to large music collections on mobile devices.

The last session consisted of three contributions dealing with specific semantic web middleware issues. The first two contributions presented a new direction in the design and development of the JBowl library for support of information retrieval and text mining. This java library and its various components and techniques have already been introduced in several of the previous WDA's. The contribution presented by Peter Butka has shown what changes in the JBowl have been done in order to move to a distributed, service-based organization of various text mining tasks. Next contribution, delivered by a graduate student Tomáš Drenčák, has focused on the possibilities how one can semi-automatically compose workflows for text mining, making use of the service version of JBowl library. Last presentation has been performed by a finishing PhD. Student Martin Sarnovsky, who presented service-oriented middleware architecture, designed and used in the EU-funded project Hydra.

Finally, we would like to thank all participants of the workshop for their contributions and very fruitful discussions after each presentation, making this year workshop a very successful event.

September 2008

Ján Paralič, Andreas Rauber

Content

Session 1: Text Classification

Session 2: Digital preservation

Session 3: Semantic web middleware

Session 1

Text Classification

Meta-learning for Automatic Selection of Algorithms for Text Classification

Gabriel Tutoky[1], Peter Bednár[1]

[1]Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Letná 9, 042 00 Košice, Slovakia
Gabriel.Tutoky@gmail.com, Peter.Bednar@tuke.sk

Abstract. This paper presents a meta-learning approach for textual document classification task and an automatic selection of the best available algorithm for creation of classifiers. After brief introductory description of principles of creation and evaluation of the classifiers, the meta-learning approach is presented as a method for automatic selection of the most appropriate classifier algorithm for creation of binary classifiers. Designed methods, based on the modification of MUDOF (Meta-learning Using Document Feature Characteristics) algorithm, are described together with its implementation using the JBowl (Java Bag of word library). Finally, the experimental results achieved by the meta-learning algorithms as well as their comparisons with traditional ways used for text classification are presented.

1 Introduction

The text classification, also sometimes referenced as the text categorization, is a method for data analysis from texts [1]. It is based on the supervised learning, where the goal is to distribute the textual documents from input data collection to the pre-defined categories. The input data collection contains a sub-set of training examples, i.e. the documents categorized in advance; these training examples are processed by statistical or machine-learning algorithms to produce the so-called classification model. The resulting model can then be applied on the rest of the input data collection to classify the textual documents without known relation to the categories.

A phase of pre-processing and text analysis is needed to identify the most relevant words, sentences, or text fragments, which have major impact to the inclusion of the text as whole to the given categories. It also affects a selection of proper classification algorithm and its settings [2].

Classification of text documents was originally designed as a semi-automatic procedure, where the users (usually experts) were responsible for selection of proper classification model, algorithms, text pre-processing methods, and optionally also to restrict the training set. In the most of application, this process of semi-automatic text classification is unusable, because users are not experts in the field of text mining. It is hard for them to select the optimal settings and the requirement was to try to

investigate the classification settings automatically, from global characteristics of the input data collection. It resulted in a design of the meta-learning method for automatic selection of classification algorithms. This method can be used in applications where is needed the classification of textual documents and where is impossible to require any optimal setting for classification process from users. The details of the meta-learning method are described in the next sections of this paper.

2 Classification, basic principles

The classification belongs to one of the basic approaches in predictive data mining. In the case of text classification, it is an approach for specific knowledge extraction from textual documents. The process of classification consists of two phases [1]:

1. *Construction of the classifier;*
2. *Usage of the classifier.*

Basic functional blocks and components used in these two phases are depicted on Figure 1.

In the *first phase*, a given set of training examples (i.e. a set of already categorized text documents) is processed to create the classifier as a model of the data behavior. In the pre-processing step, the terms are extracted from the text of documents, and the whole input set is transformed into a vector representation [2]. The vector size can be reduced by various *pre-processing* and text analysis methods as e.g. tokenization, stop-words elimination, stemming and lemmatization, term clustering (LSI), etc. [2], [3].

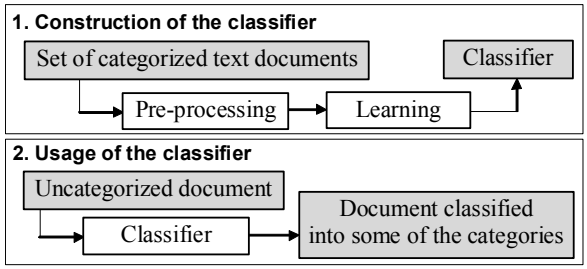


Figure 1. Two phases of the classification process

In the step of *learning*, various learning algorithms based on the statistical and heuristic techniques are used for processing the vector representation of the training set. Selection of proper algorithm and settings its optimal parameters is usually preformed manually and requires expert knowledge as well as an experience in the field of text mining. This is especially the point where the meta-learning approach (described in the next section) can help and select the most appropriate classification algorithm with respect to the characteristics of the training data set. In this paper are focused following algorithms:

- Linear classifiers: *Perceptron, Support Vector Machine (SVM)*,

- Methods based on a recursive division of the space of documents into a set of disjunctive areas: *Decision trees, Decision rules*,
- Methods based on the instances: *k-Nearest Neighbors (kNN)*.

For the construction of the classifier, it is assumed that every training example, i.e. an a-priori categorized document, belongs to one or more of the pre-defined categories c_i ($c_i \in C$, $C = \{c1, c2, ..., cN\}$, where N stands for the number of the categories). Because the document can by categorized into more than one category, the problem is decomposed into individual category level. Specifically, there is one classifier, so-called *binary classifier*, for each category. Binary classifiers are able to distinguish the documents of one category from the documents belonging to all of the rest of categories.

This way, each category can have its own binary classifier; decompose the classifier building problem and allows using different types of classifiers for various categories. The union of these binary classifiers for all categories forms the resulting classifier – so-called *classification model*, which implicitly describes the set of pre-defined categories.

The resulting classification model is used in the *second phase* for a prediction of the target categories, which are identified for the "new" (i.e. unknown, a-priori uncategorized) documents from the input data collection. The input document is processed by all the binary classifiers from the classification model and document is assigned into these categories, for which the binary classifiers has a positive value, e.g. if binary classifier CF_a classify input document as positive, than the document is assigned into category c_a. If there is no binary classifier for input document which returns the positive value, then the document is assigned as unclassified. Finally, set of categories, predicted for the input document, is generated as a result of the classification procedure.

Quality of the classification can be evaluated using the *testing data collection* of documents, which contains the documents already (a-priori) categorized into the pre-defined categories. The testing documents are classified regularly, using the produced classification model (Figure 1). The results are then compared with the a-priori categorization for each testing document. This comparison is performed by a set of statistical measures; the most frequently used indicators are the *precision*, *recall*, and combined *effectiveness measure F1*. These measures can be combined into one global measure for the space of all categories by *micro averaging* and *macro averaging* methods [2], [3]. These measures will use to evaluate the results of experiments in section 4.

3 Meta-learning

Implementation of the classification procedure in practice requires the selection of proper algorithm in the phase of classifier creation, namely in the learning step. The meta-learning approach can be used to automate the selection of the algorithms separately for each of binary classifiers, according to the specific characteristics of the training set of documents, thus resulting into more adaptive and flexible classification procedure. This approach does not require any additional effort from

user side for controlling the classification process and provides higher quality of the classification results.

The meta-learning approach is based on a design of an adaptive system, which can increase its effectiveness based on the feedback from previous "experiences", i.e. on the evaluation of the examples processed in past [4]. Selection of the best learning strategy, most suitable for particular problem, is a generalization based on accumulating experience on the performance of multiple applications, strategies, or algorithms [5]. In the domain of text classification, the meta-learning approach is able to select the most appropriate and the most effective classification algorithm according to the characteristics of the training set (as e.g. term or category distribution, average length of documents, etc.). To achieve this selection, there is a need to create the decision mechanism (meta-model) in the first step and then to use it in the second step for creation of new classifiers (cf. first phase of general text classification process, presented on Figure 1).

The process of the meta-learning approach applied in text classification for construction of classifiers consists again of the two phases, as depicted on Figure 2:

1. Construction of the meta-model;
2. Usage of the meta-model for selection of algorithms and for creation of classifiers.

First phase of the meta-model construction can be further divided into the two steps:

- Specification of feature characteristics for training documents;
- Learning of the meta-model (meta-classifier).

The feature characteristics can be obtained from the training set for each of categories and can be expressed as a vector $F_i = (f_{i1}, f_{i2}, \ldots, f_{il})$. These vectors can then be used in the step of meta-model learning for selecting the most appropriate algorithm for particular categories.

The meta-model learning is usually based on prediction of an optimization parameter, given by comparison and evaluation of the feature characteristics F_i with the values of effectiveness, i.e. with the classification errors obtained from applying pre-defined classification algorithms on the training and testing set. The meta-model can then be constructed from these values using a regression analysis or a meta-classification procedure.

Second phase of the meta-model usage is rather simple, where the feature characteristics are obtained from unknown (uncategorized) input documents and these are processed in the same way as in the phase of meta-model construction. The meta-model is then able, according to the feature characteristics of the new documents, to select and propose the most suitable classification algorithm for creation of the resulting classifier.

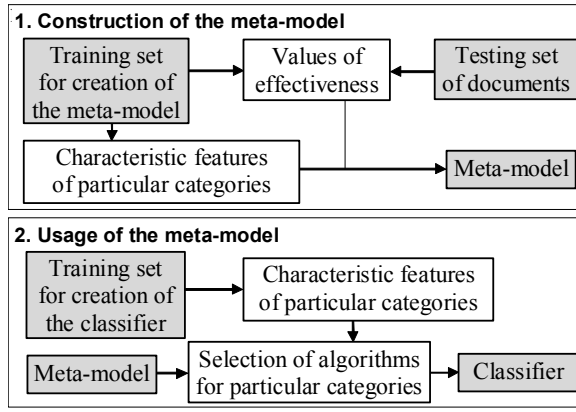


Figure 2. Meta-learning approach, two phases

In my work, I adopted the MUDOF algorithm [6], based on the multiple regression analysis of feature characteristics obtained from the training set of text documents. I have implemented the MUDOF algorithm as an extension of the JBowl library [7]. In addition, after a set of initial experiments, I have enhanced the MUDOF algorithm itself in several ways, small modification of original MUDOF algorithm and producing a new, modified version, where the meta-model learning step is based on the meta-classification procedure, using the kNN classification algorithm. The small modification of original MUDOF algorithm (signed as MUDOF_R) rests in customization of feature characteristics (read below) and in change of optimization parameter to *F1 measure* (originally classification error). The modified algorithm, referenced as MUDOF_K (i.e. MUDOF with kNN meta-learning) was also implemented into the JBowl. A set of experiments were performed to compare the effectiveness and results of the MUDOF_R (i.e. MUDOF based on regression analysis) with MUDOF_K and with the traditional classification using the five pre-defined algorithms (see section 2). Some of these experiments are described and discussed in the section 4 below.

The MUDOF proposes a set of nine (l = 9) feature characteristics [6], from which I have selected the following five:

- *AvgTopInfoGain*, average information gain of the best t terms of a given category. The information gain of individual terms is computed for current category, average is then counted from t terms with the highest information gain.
- *PosTr*, number of positive examples in the training set for given category.
- *AvgTermVal*, average weight of document's terms for given category. The average weight of terms for a single document is computed at first; then the weight is computed for all the positive examples of a given category.
- *NumInfoGainThres*, number of terms for which the information gain value exceeds a globally specified threshold.
- *AvgDocLen*, average length of a document for given category. The document's length is computed as a number of all the indexed terms in a document. The average is obtained by computing the length for all the positive examples for given category.

The selection of these feature characteristics was accomplished according to the experimental results. The above mentioned five characteristics were selected as the most representative, with the most significant influence on the selection of algorithms. In addition, the characteristics *PosTr* and *NumInfoGainThres* were modified into the form of a ratio or an average value, since it provides a more adequate description of global characteristics over particular categories. The modifications were as follows:

- *PosTr*, ratio of positive and negative examples in the training set for given category.
- *NumInfoGainThres*, ratio of the number of terms with the information gain over the threshold to the number of all the terms.

The MUDOF algorithm requires a division of the training set into two sub-sets [6]:

- training set for meta-model (*TM*),
- training set for classification model (*TC*).

The characteristic features for the categories can then be obtained from *TM* and *TC* as two separate data sets. The *TM* data are used for an estimation of the regression model parameters, and the data from *TC* are used for prediction of the optimization parameter (one of possible measures of classifier quality) for particular algorithms used within the binary classifiers for the given categories. The algorithm with the highest estimation of the optimization parameter on a category is then returned as the best (optimal) and will be used for the construction of binary classifier of this category.

The MUDOF algorithm in originally uses a prediction of the classification error for a given category, based on the characteristic features of the training documents belonging to this category. Instead of the MUDOF_R algorithm uses *F1 quality measure*, not classification error as is it used by MUDOF algorithm. The goal is modeling the relations between characteristic features and optimization parameter and to obtain the $(\hat{\beta}_{jk})$ parameters for each of the algorithms. Implementation of the MUDOF_R algorithm can be described in the steps depicted on Figure 3.

A. Meta-model construction:
Input: *TM*, *TC*, set of available classification algorithms *A*, set of categories *C*.
1. While (there is an algorithm in *A*)
2. Take an algorithm ALG_j from *A*
3. For each (category c_i from *C*)
4. Apply ALG_j on *TM* for c_i and obtain the binary classifier CF_{ij}
5. Apply CF_{ij} on *TC* for c_i and obtain the optimization parameter p_{ij}
6. End For
7. Estimate the $(\hat{\beta}_{jk})$ parameters of regression model for ALG_j using optimization parameter p_{ij} and vector of feature characteristics F_{ik} (on the *TM* training set)
8. End While

B. Usage of the meta-model:
10. For each (category c_i from *C*)
11. While (there is an algorithm in *A*)
12. Take an algorithm ALG_j from *A*
13. Estimate the optimization parameter p_{ij} using the $(\hat{\beta}_{jk})$ and corresponding F_{ik} (on the *TC* set)
14. If the p_{ij} is maximal, then the ALG_j is the best for category c_i
15. End While
16. End For

The designed modification of the MUDOF_K differs from the MUDOF_R by the meta-model learning method. Instead of linear regression, the MUDOF_K uses the classification approach, based on the kNN method. Main advantage of the proposed modification is the possibility of incremental learning of the meta-model. This feature is especially helpful in the systems, where the input data set is updated rather frequently and the changes should be reflected in the meta-model.

4 Experiments

The meta-learning algorithm MUDOF_R, based on the regression model, as well as the MUDOF_K modification, based on the *kNN* classification method, were both implemented as an extension of the *JBowl* library. The implementations were then tested in a set of experiments to prove the concept of automatic creation of classifiers by the meta-learning approach and to evaluate the quality of the resulting classification procedure.

4.1 Preparation of the testing data

The experiments were accomplished on the *Reuters-21578* [8] and *20 Newsgroups* [9] document sets. The Reuters-21578 contains 10.788 documents distributed into 90 categories. For the experiments, the document set was divided into the following subsets:

- training set (*TR*): 7.769 documents,
- testing set (*TE*): 3.019 documents.

For the meta-learning, the *TR* was further divided into the training sets for meta-model and for classifier:

- *TM*: 3.815 documents,
- *TC*: 3.961 documents.

The Reuters-21578 set is not very well balanced; it has a high variability of the documents distribution towards the categories. It contains categories with about 1.500 positive examples, as well as about 30 categories with less than 10 documents.

The 20 Newsgroups contains 19.997 documents distributed into 20 categories. The 20 Newsgroups set is well balanced and has low data variability, since almost equal number (about 1.000) of documents belongs into each of the categories. For the experiments, we have divided the 20 Newsgroups set into the following subsets:

- training set (*TR*): 10.025 documents,
- testing set (*TE*): 9.972 documents.

4.2 Experiment 1, single data set

First experiment was focused on testing of the meta-learning approach on a single data set. The goal was to prove the hypothesis that the meta-learning provides a better effectiveness and quality of the resulting classifier in comparison with the several

pre-defined classification algorithms. This experiment was performed on the Reuters-21578 document set.

The *effectiveness* of the classification was evaluated by the *F1 quality measure* mentioned in the section 2 above. The integrated measure *Macro F1*, which combines precision and recall over whole testing set, was used as the main quality measure for the experimental results. The MUDOF_K and MUDOF_R algorithms were compared with basic classification algorithms as *Decision Trees*, *Decision Rules*, *SVM*, *Perceptron*, and *kNN*. Resulting values of the quality measures are listed in Table 1, graphical comparison of the *Macro F1* measure is depicted on Figure 4. *Macro F1* has been chosen because it is the most descriptive effectiveness measure for unbalanced document sets (like Reuters-21578).

Table 1. Single data set, quality measures

Statistics	MUD-OF_K	MUD-OF_R	Dec. Trees	Dec. Rules	SVM	Perc.	kNN
Micro Precision	0,808	0,869	0,790	0,792	**0,932**	0,885	0,852
Micro Recall	**0,860**	0,820	0,793	0,801	0,785	0,794	0,792
Micro F1	0,833	0,844	0,792	0,796	**0,852**	0,837	0,821
Macro Precision	0,567	0,556	0,521	0,499	**0,580**	0,556	0,496
Macro Recall	**0,520**	0,502	0,503	0,492	0,369	0,356	0,384
Macro F1	**0,543**	0,527	0,511	0,495	0,451	0,434	0,433

The results demonstrate that the MUDOF algorithms, using the meta-learning approach, are able to provide higher values of the resulting effectiveness, expressed by the Macro F1 measure. For the macro measure, the MUDOF has similar results as Decision Trees and Rules. However, the MUDOF has better results for the Micro measure. The SVM, Perceptron, and kNN have similar and slightly better (in case of SVM) results as MUDOF for the Micro measures, but the MUDOF is better in the results for Macro measures. Percentage increase of the MUDOF algorithms in comparison with the globally best basic algorithm, i.e. Decision Trees, was 4,1% for the Micro F1 and 3,1% for the Macro F1 measure.

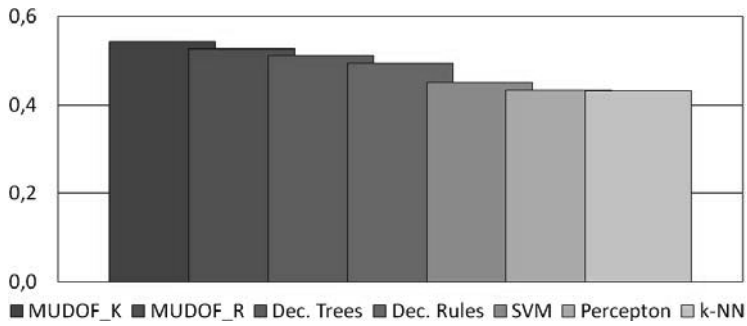


Figure 4. Single data set, comparison of the Macro F1 measure for algorithms

4.3 Experiment 2, two data sets

In the second experiment, the goal was to test the usability of the meta-learning approach on two different sets of documents, achieving a situation where meta-model has been trained on different dataset than it has been tested later on. Meta-model learning phase was performed on the Reuters-21578 document set (we have used the same meta-model as in previous example), and the resulting classifier was constructed on the 20 Newsgroups data. The process of obtaining results and their evaluation is the same as in the first experiment. Values of the effectiveness measures for the experiment with two data sets are presented in Table 3, graphical comparison of the Macro F1 measure is depicted on Figure 5.

It follows from the achieved results that the *SVM* algorithm is the best for the balanced data of the 20 Newsgroups. All the algorithms except *Perceptron* have almost equal results, no significant improvement was achieved by applying the meta-learning (Figure 5). In the case of balanced data, a single algorithm can be selected as the best – in our case it is the *SVM*. The meta-learning approach is able to assure that the resulting effectiveness will be "close" to the best, and avoid a selection of the algorithms with bad effectiveness (Perceptron, in our case).

Table 3. Two data sets, quality measures

Statistics	MUD-OF_K	MUD-OF_R	Dec. Trees	Dec. Rules	SVM	Perc.	kNN
Micro Precision	0,824	0,899	0,892	0,892	**0,961**	0,286	0,838
Micro Recall	**0,894**	0,871	0,873	0,873	0,843	0,383	0,847
Micro F1	0,857	0,884	0,883	0,883	**0,898**	0,328	0,843
Macro Precision	0,830	0,896	0,891	0,891	**0,958**	0,782	0,845
Macro Recall	**0,895**	0,869	0,875	0,875	0,844	0,384	0,848
Macro F1	0,861	0,882	0,883	0,883	**0,897**	0,515	0,846

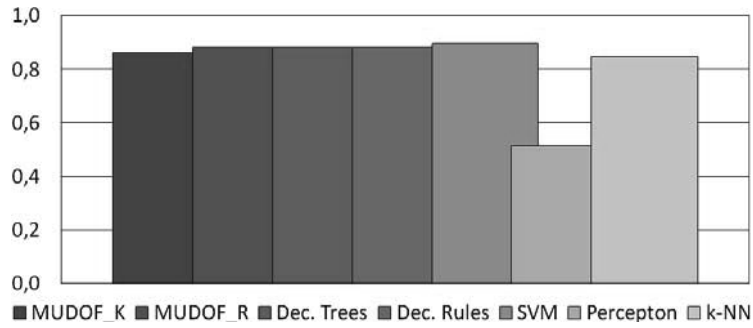


Figure 5. Two data sets, comparison of the Macro F1 measure for algorithms

5 Conclusions

The presented meta-learning approach towards the text classification seems to be a suitable method for support of automatic classification in user-oriented systems. The original MUDOF meta-learning algorithm, based on the linear regression, was modified and adapted using the kNN classification method for meta-model creation. Both algorithms were tested on the Reuters-21578 and 20 Newsgroups document sets and the results indicate that the meta-learning increases effectiveness and quality of the results. However, as is shown in the case of balanced training set (Experiment 2), there is still some space for further improvements of meta-learning algorithms. After all, the proposed meta-learning approach can be considered as a technology, which enables automatic and adaptive text classification, increases quality of the classification results, and can be effectively used in the user-oriented systems in practice.

Acknowledgment

The work presented in this paper was supported by: the Slovak Research and Development Agency under the contract No. RPEU-0011-06 and No. APVV-0391-06; the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic under grant No. 1/4074/07 and MVTS project No. Fr/ČR/SR/TUKE07.

References

1. Paralič, J.: Knowledge Discovery in databases and texts, Habilitation thesis, Technical University of Kosice, Slovakia, 2003.
2. Bednár P.: Authomatic classification of texts based on the content (in Slovak), Concept of PhD. thesis, TU Košice, Slovakia, 2004.
3. Sebastiani F.: Machine Learning in Automated Text Categorization, ACM Computing Surveys, Vol. 34, Iss. 1, New York, USA, 2002, pp. 1-47.
4. Vilalta R., Drissi Y.: A Perspective View And Survey Of Meta-learning, AI Review, Vol. 14, No. 2, Springer Netherlands, 2002, pp. 77-95.
5. Vilalta R., Giraud-Carrier Ch., Brazdil P.: Meta-Learning: Concepts and Techniques, The Data Mining and Knowledge Discovery Handbook, Springer US, 2005, pp. 731-748.
6. Wai L., Kwok-Yin L.: A meta-learning approach for text categorization, Proc. of the 24th ACM SIGIR conference, New Orleans, USA, 2001, pp. 303-309.
7. Bednár P.: JBowl, Java Bag of word library, available at http://sourceforge .net/projects/jbowl/, Accessed: 12th May 2008.
8. Lewis D.: Test data Collection Reuters-21578, available at http://www. Daviddlewis.com/resources/testcollections/reuters21578/, Accessed: 12th May 2008.
9. The UCI KDD Archive: 20 Newsgroups, University of California, Irvine, 1999, available at http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups .html, Accessed: 12th May 2008.

Locally Informed Methods for Text Classification

Peter Smatana

Technical University of Košice,
Faculty of Electrical Engineering and Informatics,
Department of Cybernetics and Artificial Intelligence,
Letná 9, Košice, 040 01, Slovak Republic
Peter.Smatana@tuke.sk

Abstract. There are some different approaches for textual document analysis. Most of them are based on statistical analysis. Statistical based approaches are very useful and important but the main problem is that they are using term reduction methods which cause losing of information in represented textual document. Our goal is to improve classification methods to decrease of amount of lost information. We can divide document to smaller parts which will be represented individually and in the process of statistical analysis individual block will represent information covered by that block instead whole document representation in traditional approach. This paper presents few different approaches how to growth quality of classification.

1 Motivation

Text classification problem is still actual area for the research. That field is fed by data from many CMS systems which have to annotate uploaded documents and it is helpful to recommend users some annotation categories. There are lots of others fields where classification can be helpful. Basic motivation point was how to minimize of losing of information without increasing of processing complexity. Next chapters will show you how to use distributional information for better text classification.

2 Background

In text classification, we are given a description $d \in X$ of a document, where X is the document space; and a fixed set of classes $C = \{c_1, c_2, ..., c_j\}$. Using a learning method or learning algorithm, we then wish to learn a classifier or classification function γ that maps documents to classes:[1]

$$\gamma = X \rightarrow C$$

The basic schema for solving classification problem of textual document is shown at the Fig.1. That schema shows few steps how to achieve assigning of the class to

processed document. Document is typically unstructured (or semi-structured) textual content.

Step of preprocessing is the most important and language dependent block which prepare text for representation block. There could be full linguistic analysis (morphology analysis, syntactic analysis, semantic analysis, etc.) which can be described as lossless information preprocessing step or there could be used losing information preprocessing methods (stemming, stop words, etc.). Problem of time complexity and vocabulary dependent preprocessing algorithms cause that those text classification algorithms using just losing information preprocessing methods. After document preprocessing step we are going to represent document for classification algorithm. There are different types of representations: term document matrix, n-grams, LSI, etc.

Last step is used for classification. There are lots of Machine Learning techniques as SVM (the top classifier for text documents [1]), KNN, Decision Trees, etc.

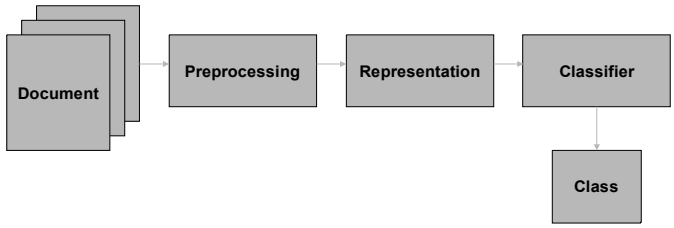


Fig. 1. Basic schema of statistical approach to text classification

Fig.2. shows that traditional approach (on the top) put all information to one bag; on the bottom figure you can see that different chapters are focused to different topics. This should be additional information for classifier. Therefore it should be improving factor for better classification using distributional information about terms.

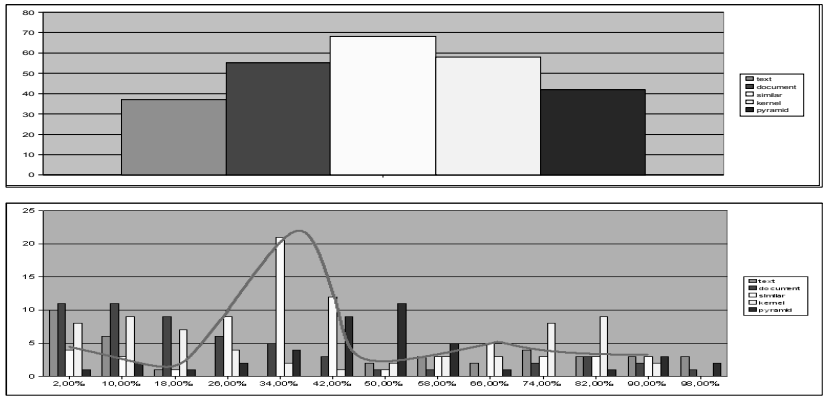


Fig. 2. Distribution of some terms in document should be self describing factor for text classification (occurrence of the terms in whole document (top), occurrence of the terms in individual parts of the document (bottom))

3 Locally Informed Methods Overview

There are lots of different approaches how to improve text classification. We can improve preprocessing, representation or classification algorithm. The most important thing is to beware of losing information at the beginning of the classification chain. That means good preprocessing and good representation can be beginning for good classification.

We would like to show you some representations techniques and how to add some additional information to representation. Problem of representation is closely connected to preprocessing problem therefore sometimes you have to use additional preprocessing techniques.

Paper "Distributional features for text classification" [2] presents very interesting approach that use position of first occurrence of specific term and the compactness of the appearances of a it as additional features in representation.

"Local word bag model" [4, 5] is based on conventional BOW model (it ignores the detailed local text information, i.e. the co-occurrence pattern of words at sentence or paragraph level). This approach represents a document as a set of local tf-idf vectors which are used for measuring similarity of documents.

Latent Semantic Indexing (LSI) has been shown to be extremely useful in information retrieval, but it is not an optimal representation for text classification. Method called "Local Relevancy Weighed LSI" improving text classification by performing a separate Single Value Decomposition.

Interesting method for recommendation books to readers in order to their favorite book is web portal BookLamp[1]. The main algorithm is based on different points of view (Density, Action, Pacing, Description, Dialog etc.) on the same book. The representation of each point of view consists from the vector representation through whole document split to section. Recommending of the book is based on comparing of these points of view.

4 Design and Implementation

Sparsity is the major problem in the statistical text representation methods. That problem grows up in case of splitting document to smaller segments. We would like to solve problem of sparsity by using Gaussian function for weighting of term occurrence in part of the documents. It means that we will represent whole document when we would like to represent just specific segment of it but term which occurs in that segment will have higher value of weight as shown at Fig.3.

[1] http://booklamp.org/

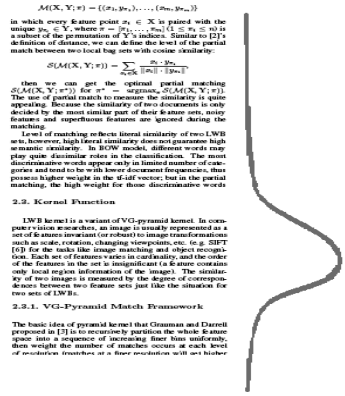


Fig. 3. Locally weighted term indexing

The JBowl Java library [7] was taken as an implementation platform for the improved representation methods for text classification. This open source software package was developed to support information retrieval and text mining tasks. The library is built on the modular framework architecture, which is highly extensible and supports SOA principles. Since it offers the required functionality for pre-processing, indexing and further exploration of text collections, it was chosen as a good candidate for implementing the classification tasks (see Fig.4.) [8].

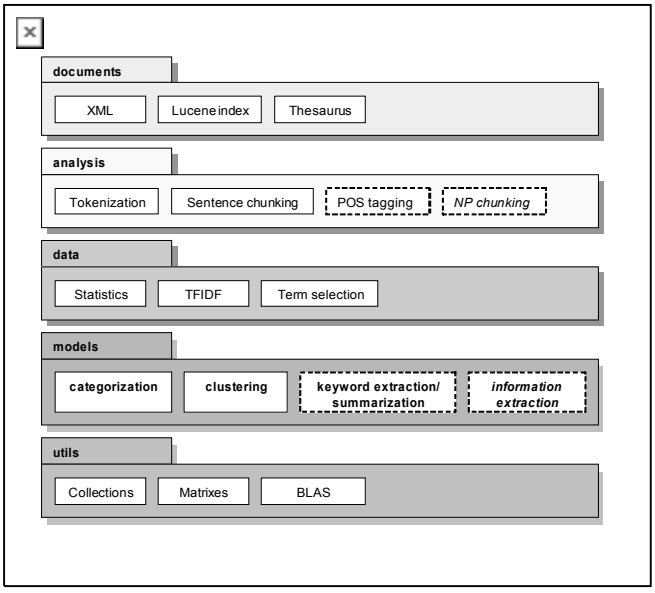


Fig. 4. JBowl library – platform for implementation of the method

5 Conclusion

There are lots of different approaches how to increase quality of results of classification task. It should be based on improving of a classification algorithm or based on a lot of other improving techniques but the most important point is to increase of quality of representation of textual documents. We could decrease of losing of information consisted in textual document by presented methods in previous chapters but there arise problem with sparsity of representation in term-document matrix. We would like to deal with that problem by the method presented in ”Description and implementation” chapter.

These local informed algorithms still lose lot of information from textual document but it is compromise between losing information and computing complexity. The best approach for lossless representation and then categorization is using chain of natural language processing (NLP) techniques where the result of preprocessing is fully semantic representation of textual document. That approach is dependent on language of the documents because all preprocessing blocks are built on language specific characteristics. Therefore locally informed methods should be compromise between NLP methods and methods which use statistical approach over whole document.

Acknowledgement

The work presented in the paper is supported by the Slovak Research and Development Agency under the contract No. RPEU-0011-06 (project PoZnaŤ) and by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/4074/07 and No. 1/3135/06; and MVTS project No. Fr/ČR/SR/TUKE07.

References

1. Manning Ch. D., Raghavan P., Schütze H.: Introduction to Information Retrieval, 2008 Cambridge University Press, ISBN 0-521-86571-9
2. Xiao-Bing Xue, Zhi-Hua Zhou: Distributional Features for Text Categorization, IEEE Transactions on Knowledge and Data Engineering, 31 July 2008.
3. Tao Liu, Zheng Chen, Benyu Zhang, Wei-ying Ma, Gongyi Wu: Improving Text Classification using Local Latent Semantic Indexing, icdm, pp. 162-169, Fourth IEEE International Conference on Data Mining (ICDM'04), 2004
4. Wen Pu, Ning Liu, Shuicheng Yan, Jun Yan, Kunqing Xie, Zheng Chen: Local Word Bag Model for Text Categorization, icdm,pp.625-630, 2007 Seventh IEEE International Conference on Data Mining, 2007
5. Lebanon G., Mao Y., Dillon J.: The Locally Weighted Bag of Words Framework for Document Representation. Journal of Machine Learning Research 8 (2007)
6. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press New York /Addison-Wesley, 1999. ISBN 0-201-39829-X
7. Bednár, P., Butka, P. (2005): JBOWL - Java Bag-Of-Words Library. In: 5th PhD student conference and scientific and technical competition of students of FEI TU

Košice, Proc. from conference and competition, Košice, Slovakia, ISBN 80-969224-4-0 (2005) 19-20

8. P. Smrž, J. Paralič, P. Smatana, K. Furdík: Text Mining Services for Trialogical Learning, In: P. Mikulecký, J. Dvorský, M. Krátký (eds.), Proceedings of the Czech-Slovak scientific conference Znalosti (Knowledge) 2007, Ostrava, Česká republika, február 2007, s. 97 - 108, ISBN 978-80-248-1279-3.

Initial Stages of Medical Text Processing Applications Set

Pavol Jasem Jr.[1], Marek Dudáš[2], Saskia Dolinská[2], Ján Paralič[1]

[1] Technical University of Košice, Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Slovakia
[2] Pavol Jozef Šafárik University in Košice, Institute of Biological and Ecological Sciences, Faculty of Sciences, Slovakia
[1]{Pavol.Jasem.Jr, Jan.Paralic}@tuke.sk, [2]{Marek.Dudas, Saskia.Dolinska}@upjs.sk

Abstract. Online freely accessible sources of information on medical topics are rapidly growing in volume. There has arisen a need for a system joining these sources and also for a system providing easier navigation in data and searching in them. This system has been developed and it is improving in its searching capabilities, and also keeps extending the number of databases, which data it handles.

Keywords: disease, defect, syndrome, congenital, hereditary, OMIM.

1 Introduction

Biomedical knowledge usually reaches the end users with a considerable lag behind the newest published discoveries. Even if we manage to collect texts of new experimental studies or clinical experiments on any particular topic, the data quantity usually exceeds the human capacity to process the data in a reasonable time. The problem is more prominent in the case of clinical genetics - sometimes we need information on a heart defect for one patient, an hour later about a kidney malformation for another patient etc., while preparation of targeted recherché with the help of the most modern bibliographic tools takes from days to weeks. Thus, patients do not get a diagnostic care on the highest achievable level even in the most developed countries in the world. The major prominent genetic databases are e.g.: OMIM (Online Mendelian Inheritance in Man), MGI, GenBank, Entrez Nucleotide, Entrez Genome, Gene Ontology, Sanger Center, EOL, EnsEMBL. None of the named databases supports synergistic collaboration with any other named database. This article describes a system that improves an information retrieval among data provided by biomedical database NCBI (National center for biotechnology information).

2 NCBI databases and web service analysis

For analysis of a database structure and following implementation of the application for the first stages of Gemin project, data in NCBI database have been

selected, for their high information volume (as for the number of articles, or large structure the data are saved in).

NCBI Entrez is one of the most authoritative sources of biomedical information. The present form of Entrez is an extremely valuable base of data in the widest sense, but it's data-access structure presented to users is not a real database or datawarehouse in the usual meaning. Rather, it is a hybrid of text entries with limited elements of database organization. In addition, individual Entrez databases are functionally different from each other.

Subsystems OMIM and OMIA (Online Mendelian Inheritance in Man / in Animals), containing the information on hereditary disorders, strike examples of the data problems of the Entrez system. Individual entries (descriptions of individual illnesses) have a form of structured text, but there are large inconsistencies in classification of individual entries and their (in)completeness. Multiple switches in the source tagged text (ASN.1, XML) point to absent text sections and there were developed difficult systems of inline text descriptors that fix the presence of different informational exceptions. For physicians or biologists in a common daily practice, the system Entrez OMIM appears to be a cumbersome and uneasy searchable text catalogue of Mendelian disorders, far from real practical needs and isolated from all Entrez-unrelated information sources. All other data sources, taking in account their smaller size and weak interconnectivity, encounter even more prominent limitations, but they often contain multiple unique pieces of information.

Data accessible through NCBI web interface don't indicate any structure, mostly for less advanced users, who filter articles by simply entering a search string into textbox and selecting a database where the search should be processed. This search is purely full-text and doesn't consider the structure NCBI has the data stored in. However, most of records in databases on the server have attributes called "Links." These attributes contain a string made up of numbers, IDs referring to articles related to the original article. This was one of reasons why a database structure of relational database was proposed, where individual NCBI databases are represented as tables. This approach enables us to search (filter) data also by the links. It is possible to search results by other attributes of the article as well (e.g. gene/protein/syndrome symbol, locus, article type in OMIM table [1]).

3 Developed subsystems

Genetics is fast-developing scientific field, and the most of information is saved on internet. However, access to these data is insufficient to the scientists, and search in such data is very restricted. Electronic solution of sophisticated clinical or scientific questions is impossible. Moreover, these web resources do not offer any other language than English, which is very discouraging for most of the "common" medical practitioners, worldwide. Therefore, it is necessary to design some structure allowing the user to form more specific queries to filter the data. Proposed system consists of more cross-communicating parts (subsystems). Although they use the same data, these subsystems have been created to be as much independent as possible, to be easily extended without interfering with other parts.

The first part of the system for automated data retrieval is a relational database, designed and implemented to serve as a primary storage for data acquired by Entrez web service. The second part is an application acquiring these data from NCBI database and saving them to the proposed database structure. These data can be accessed either by a web application or a web service, which comprise the third part of the system. The web service provides data access by whole table contents printing, or other records based on specified database query. Although the web application can provide an interface friendlier to a wider spectrum of users, with lesser emphasis on user's computer skills, it enables the most effective access to the data, proposing the main contribution of this work.

3.1 Relational database for NCBI data with tables for reorganized data

Structure of local database was derived from structure of NCBI databases provided by Entrez Utilities. However, with respect to database normal forms, some attributes has been modified. E.g. links to records in another database: these links are provided as one string containing numbers delimited by comma. These are transformed to multiple rows in table of links.

Data in many tables are now copied and reorganized to new tables, with a new structure, keeping the shadow copy of NCBI in original tables. This includes for example a table containing clinical synopses. Each clinical synopsis (CS) in NCBI web service consists either of a CS key, or a CS key with detailed description. Following is not evident from retrieved data, but: a CS with only a key is a category (gross location of a CS). All subsequent CS key-detail pairs should be child nodes of that (first) category. The key in a pair stands for a subcategory of its detail (brief description of defect). Thus, data from tables regarding clinical synopses are being slightly modified (corrected typographical errors) and saved to tables representing tree topology of all observed defects with its locations within human body. Those categories are re-saved as revised, to reflect logical category-subcategory relations. For example, category “Head, skin and hair” should be same as “Head, hair, skin”, or, “Head” category should be child to this category.

Even though GO data are not compatible with MS SQL server, the developed database also possesses data from Gene ontology, which haven’t been used in web application yet. Those data are periodically downloaded and updated.

3.2 System for automatic information retrieval as a Windows service

Data from NCBI are copied into the local database through a system for automated retrieval of data from NCBI. It uses Entrez Utilities [2] and NCBI web service for downloading the data. The developed application (named Entrez Downloader) searches for records on NCBI server by entered search strings saved in local database and stores the data into the database.

These search strings are formulated to match all records present in NCBI database. This application was later changed to a combination of a Windows application and a system service, being managed by the application. Since this upgrade, there’s one less

duty for a server's administrator (administrator of the operating system, where Entrez Downloader is running). The service is running without a need of logging to the system GUI after every system restart.

3.3 End-user interfaces

A web service and a web application were developed for enabling user to access the retrieved data. Individual functions of the web service are separated into two classes (thus there are practically two web services). Methods in the first class can be used to select data from database tables, related to data acquired from NCBI databases, presuming entrance of any SQL command on data selection. The second class is used to manage web application user accounts. Developed web application is used to provide a friendly user interface to filter or browse downloaded data, and to enter search strings for Entrez Downloader.

4 Results of designed systems usage

We stated several questions, which the system should be able to give answers for. For all records found by full-text search on NCBI server, a user is able to filter records by 137 attributes (number of attributes in database OMIM, relevant for a scientist, is 62, in Nucleotide and Protein databases there are 51 attributes, and 24 ones in OMIA database). Developed web application offers an ability to filter downloaded data by some of the attributes, and the web service allows filtering by all of them. An example question is stated below, which a scientist can get answer for, using our system.

"Which diseases are related to upper limbs and cleft palate regardless of a gender, and without regard to whether a gene causing the disease is known or not?"

A user gets the answer after entering "cleft palate" string as a full-text search in web application and three-mouse-click setting in the filter. The query returned 23 records, and 15 of them were relevant for most specialized scientists.

4.1 Further applications

Downloaded and revised data allows us to examine data and text by several approaches, such as classification, clustering, data and text mining, etc. A simple clustering was done using GHSOM but we got irrelevant categories (clusters) and we found 1-to-1 classification insufficient (more clusters needed for one article).

An application for PubMed articles automated downloading is developed, which will later include a system with detailed searching algorithms, and downloaded documents will be used for text mining and citation linking.

Simple classification (kNN algorithm) has been performed over PubMed articles bringing out a result of 90.5% classification success rate when classifying cleft-related and not related diseases in 1419 OMIM results (compared to manual categorization).

5 Creating relations between datasets

Several authors needed to compare data from various publicly available databases. If these data were stored in one database, their comparison would be much easier. Moreover, we would not only be able to simply compare results retrieved from more databases, but we could even join these multiple data and get results from running a study on this wider spectrum of information. In this way, many additional facts can be taken into consideration right in one of the first stages of data collecting process. For example, some of the results retrieved in [3] by Becquet, et al., using strong-association-rule mining on human SAGE data [4], were compared to NCBI data.

Gene ontology (GO) is one of the most important sources of information that should be included in such database (the contribution is described, for example, in [5]). Suitable relational representation of information present in GO will help users to display the data in database, and will also help them to navigate through them in a way that is best known for scientists.

Creating relations between more datasets will provide more exact source of information for application of additional data- or text-mining techniques.

Acknowledgements

The work presented in this paper is supported by “Information Extraction and Knowledge Discovery for Support of Hereditary Diseases Research (Eco-Net)” International Research Cooperation project (Fr/ČR/SR/TUKE07, “Genomic Data Mining on Birth Defects (GEMIN)” Slovak Ministry of Health project (2007/65/UPJŠ-02) and “Support for Knowledge Creation Processes” Slovak Research and Development Agency project (RPEU-0011-06).

References

[1] OMIM database. NCBI HomePage. [Online] NCBI. [Accessed: April 23, 2007.] http://www.ncbi.nlm.nih.gov/omim.

[2] NCBI Web Service. NCBI HomePage. [Online] NCBI, February 2, 2007. [Accessed: April 24, 2007.] http://eutils.ncbi.nlm.nih.gov/entrez/query/static/esoap_help.html.

[3] C. Becquet, S. Blachon, B. Jeudy, J.F. Boulicaut, O. Gandrillon, "Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data," Genome Biol, vol. 3, no. 12, 2002.

[4] V.E. Velculescu, L. Zhang, B. Vogelstein, K.W. Kinzler, "Serial analysis of gene expression," Science, vol. 270, pp. 484-487, October 1995.

[5] J. Kléma, A. Soulet, B. Crémilleux, S. Blachon, O. Gandrillon, "Mining Plausible Patterns from Genomic Data." In Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems (June 22 - 23, 2006). CBMS. IEEE Computer Society, Washington, DC, pp. 183-190, 2006.

Session 2

Digital preservation

Automating the preservation planning process: An extensible evaluation framework for digital preservation

Christoph Becker

Vienna University of Technology, Vienna, Austria
http://www.ifs.tuwien.ac.at/~becker
{becker}@ifs.tuwien.ac.at

Abstract. The dominance of digital objects in today's information landscape has changed the way humankind creates and exchanges information. However, it has also brought an entirely new problem: the longevity of digital objects. Due to the fast changes in technologies, digital documents have a short lifespan before they become obsolete. Digital preservation, i.e. actions to ensure longevity of digital information, thus has become a pressing challenge. Different strategies such as migration and emulation have been proposed; however, the decision between available strategies and the evaluation of potential tools is very complex. Preservation planning supports decision makers in reaching accountable decisions by evaluating potential strategies against well-defined requirements. The analysis of the qualitative and quantitative performance of different migration tools for digital preservation has to rely on validating the converted objects and thus on an analysis of the logical structure and the content of documents.
Different approaches exist for analysing and characterising digital objects. However, the connection to the specific requirements and criteria that have to be considered in the evaluation procedure is yet unclear, and there is no automated and traceable way of linking these characteristics to the decision factors. Furthermore, an integration of preservation action, characterisation and planning is missing.
This paper presents an extensible integration architecture for automating the analysis and evaluation of potential preservation actions. We describe the problem context and the planning methodology underlying the approach. We then present an overall integration architecture and an extensible evaluation framework connecting requirements and criteria to measurable factors both in the environment and the digital objects themselves. We discuss the problems and expected benefits and outline the next steps towards implementing the proposed solution.

1 Introduction

The last decades have made digital objects the primary medium to create, shape, and exchange information. An increasing part of our cultural and scientific heritage is being created and maintained in digital form; digital content is at the heart of today's economy, and its ubiquity is increasingly shaping private lives.

The ever-growing complexity and heterogeneity of digital file formats together with rapid changes in underlying technologies have posed extreme challenges to the longevity of information. So far, digital objects are inherently ephemeral. Memory institutions such as national libraries and archives were amongst the first to approach the problem of ensuring long-term access to digital objects when the original software or hardware to interpret them correctly becomes unavailable [27].

A variety of tools performing preservation actions such as migration or emulation exist today; most often, there is no optimal solution. The complex situations and requirements that need to be considered when deciding which solution is best suited for a given collection of objects mean that this decision is a complex task. This multi-criteria decision making process is one of the key issues in preservation planning. Preservation planning aids has to evaluate available solutions against clearly defined and measurable criteria. This evaluation needs verification and comparison of documents and objects before and after migration to be able to judge migration quality in terms of defined requirements. It thus has to rely on an analysis of the logical structure of documents that is able to decompose documents and describe their content in an abstract form, independent of the file format. Moreover, there are a number of other factors to take into account, such as risk factors of object formats, cost models that influence the planning decisions, and changing constraints in environments and usage that imply a change in preferences and/or requirements.

Creating and maintaining the conceptual connection between these influence factors and the outcomes of decisions is a difficult process and a largely unsolved question. The effort needed to analyse objects, requirements and contextual influence factors is in many cases prohibitive. Furthermore, the evaluation of alternative action paths to arrive at accountable recommendations for a preservation action component is often costly.

This paper presents an evaluation framework that aims at automating this process and improving the traceability of influence factors in digital preservation decision making. An integration architecture brings together preservation planning with preservation actions, characterisation services, and the heterogeneous information sources and registries where these are described.

The article is structured as follows. We describe the context of work in the next section and the preservation planning methodology which forms the basis of our approach in Section 3. Section 4 presents the planning tool Plato which forms the technical background for implementing the described framework. Section 5 then outlines the integration architecture, while the last section discusses potential benefits and issues and outlines future work.

2 Related Work

Digital preservation is a pressing matter – large parts of our cultural, scientific, and artistic heritage are exposed to the risks of obsolescence. Trustworthiness is probably the most fundamental requirement that a digital repository preserving

content over the long term has to meet. The Trusted Repository Audit and Certification Criteria as a widely recognised step towards standardisation and certification of digital repositories define a set of requirements to be followed in digital preservation processes [26]. At the heart of a preservation endeavour lies preservation planning. It is a core entity in the ISO Reference Model for an Open Archival Information System (OAIS) [12].

The rising awareness of the urgency to deal with the obsolescence that digital material is facing has led to a number of research initiatives over the last decade. Research has mainly focussed on two predominant strategies – migration[25, 16] and emulation[20, 28]. Migration, the conversion of a digital object to another representation, is the most widely applied solution for standard object types such as electronic documents or images. The critical problem generally is how to ensure consistency and authenticity and preserve all the essential features and the conceptual characteristics of the original object whilst transforming its logical representation. Lawrence et. al. presented different kinds of risks for a migration project [15].

In contrast to migration, emulation operates on environments for objects rather than the objects themselves. Emulation aims at mimicking a certain environment that a digital object needs, e.g. a certain processor or a certain operating system. Rothenberg [20] envisions a framework of an ideal preservation surrounding for emulation. Recently, Van der Hoeven presented an emerging approach to emulation called *Modular emulation* in [28].

In principle, the selection problem in digital preservation can be seen as a domain-specific instance of the general problem of Commercial-off-the-Shelf (COTS) component selection [19]. The field of COTS component selection has received considerable attention in the area of Software Engineering. A comprehensive overview and comparison of methods is given in [17]. One of the first selection methods presented was the Off-the-Shelf-Option (OTSO) [13, 14]. It provides a repeatable process for evaluating, selecting and implementing reusable software components. OTSO relies on the Analytic Hierarchy Process (AHP) [22] to facilitate evaluation against hierarchically defined criteria through series of pairwise comparisons. Other methods include CRE[1] and PORE[18]. Most selection methods follow a goal-oriented approach [29] and conform to what Mohamed calls a 'General COTS selection process (GCS)' [17], an abstract procedure with the steps *Define criteria, Search for products, Create shortlist, Evaluate candidates, Analyze data and select product.*

The PLANETS preservation planning methodology[23] defines measurable requirements for preservation strategies in a hierarchical form and evaluates them in a standardised setting to arrive at a recommendation for a solution. The procedure is independent of the solutions considered; it can be applied for any class of strategy, be it migration, emulation or different approaches, and has been validated in a series of case studies [4, 7, **?**]. An OAIS-based analysis of the approach is shown in [24].

An important aspect of the evaluation process is the need for automatic validation and comparison of objects. A number of tools and services have been

developed that perform content characterisation specifically for digital preservation. The National Library of New Zealand Metadata Extraction Tool[1] extracts preservation metadata for various input file formats. Harvard University Library's tool JHove[2] enables the identification and characterisation of digital objects. Collection profiling services build upon characterisation tools and registries such as PRONOM[3] to create profiles of repository collections [8]. The eXtensible Characterisation Languages presented in [6] support the automatic validation of document conversions and the evaluation of migration quality through a analysis and decomposition of digital objects into their constituting elements, thus representing them in hierarchical form in an abstract XML language.

Some approaches deal with distributed preservation architectures. Hunter [11] describes a distributed architecture for preserving composite digital objects using ontologies and web services. Ferreira [10] presents a system for performing format migrations based on pre-specified requirements.

The EU project 'Preservation and Long-Term Access via Networked Services' (PLANETS)[4] is creating a distributed service-oriented architecture as well as practical services and tools for digital preservation [9]. Based on a common conceptual framework, it is developing services for preservation action, characterisation, testing and planning.

3 The preservation planning workflow

The Planets preservation planning workflow as described in [23] consists of three main stages:

1. **Requirements definition** is the natural first step in the planning procedure, collecting requirements from the wide range of stakeholders and influence factors that have to be considered for a given institutional setting. This includes the involvement of curators and domain experts as well as IT administrators and consumers. Requirements are specified in a quantifiable way, starting at high-level objectives and breaking them down into measurable criteria, thus creating an *objective tree* which forms the basis of the evaluation of alternative strategies. Furthermore, as this evaluation would be infeasible on the potentially very large collection of objects, the planner selects representative sample objects that should cover the range of essential characteristics present in the collection at hand.
   While the resulting objective trees usually differ through changing preservation settings, some general principles can be observed. At the top level, the objectives can usually be organised into four main categories:
   - *Object characteristics* describe the visual and contextual experience a user has by dealing with a digital record. These characteristics are often referred to as *significant properties*. Subdivisions may be "Content",

[1] `http://meta-extractor.sourceforge.net/`
[2] `http://hul.harvard.edu/jhove`
[3] `http://www.nationalarchives.gov.uk/pronom`
[4] `http://www.planets-project.eu`

"Context", "Structure", "Appearance", and "Behaviour" [21], with lowest level objectives being e.g. the preservation of color depth, image resolution, forms of interactivity, macro support, or embedded metadata.
- *Record characteristics* describe the technical foundations of a digital record, the context, interrelationships and metadata.
- *Process characteristics* describe the preservation process. These include usability, complexity or scalability.
- *Costs* have a significant influence on the choice of a preservation solution.

The objective tree documents the individual preservation requirements of an institution for a given partially homogeneous collection of objects. An essential step is the assignment of measurable effects to the objectives. Wherever possible, these effects should be objectively measurable (e.g. € per year, frames per second). In some cases, such as degrees of openness and stability or support of a standard, (semi-) subjective scales will need to be employed. Strodl et. al. [23] report on a series of case studies and describe objective trees created in these.

2. The **evaluation of potential strategies** is carried out empirically by applying selected tools to the defined sample content and evaluating the outcomes against the specified requirements.
3. **Analysis of the results** takes into account the different weighting of requirements and allows the planner to arrive at a well-informed recommendation for a solution to adopt.
4. The final phase of **preservation plan definition** then uses the documented recommendation to define a concrete action plan for preserving the given set of digital objects[5].

The described workflow provides a solid, well-documented and well-tested approach of empirically evaluating potential solutions and defining concrete action steps. However, it is of considerable complexity and requires substantial effort, if not properly supported by according software. Curators and preservation planners do not have enough information and options at hand, they do not know potential strategies and are unsure how to model, quantify and measure their requirements. Morevoer, they find it very difficult to establish the complex relationships between technical, domain specific, and contextual influence factors and the impact they have on the decisions. Furthremore, researchers and tool developers are lacking a common framework of delivering, deploying, testing, distributing and putting to use algorithms and tools for analysis and characterisation, i.e. the analysis and comparison of object properties and the characterisation of actions.

Thus an automated platform is needed to support measurements, evaluation, and decision making. The planning tool Plato whose vision was described in [5] strives to provide full support for preservation planning endevaours following the described approach. It is a web-based software tool that guides the preservation planner through the workflow.

This paper describes how a pluggable evaluation architecture can be leveraged to integrate both information from diverse sources and services for preservation action and characterisation. We describe the overall integration architecture

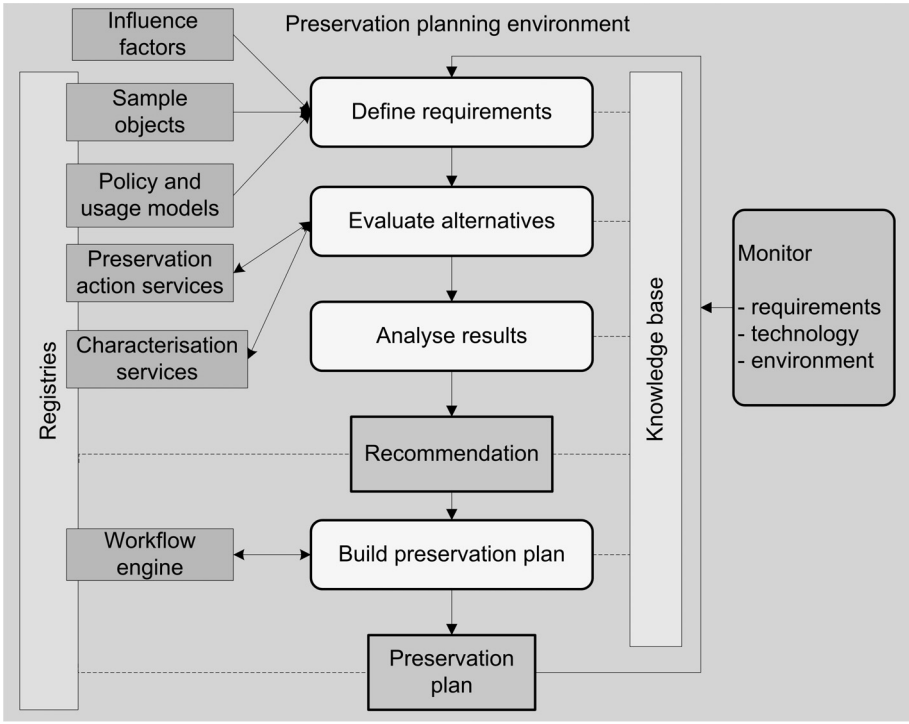


Fig. 1. Preservation planning environment

and then focus on automating the evaluation of requirements and criteria as they are defined in the objective tree. To this end, we propose a pluggable framework relying on *modellers, evaluators and comparators* to connect requirements and criteria to measurable and traceable properties and thus automate the planning procedure. Additionally, *watchers* are foreseen for continously monitoring environmental factors and constraints.

The following section describes the context of work by presenting the planning tool Plato and some of the existing and emerging services that need to be leveraged and connected to the evaluation procedure. We then describe the integration architecture in Section 5 and provide an outlook to future work in Section 6.

4 The planning tool Plato

The planning tool implements the preservation planning workflow described above and includes additional external services to automate the process. The software itself is a J2EE web application relying on open frameworks such as Java Server Faces and AJAX for the presentation layer and Enterprise Java Beans for the backend. It is integrated in an interoperability framework that

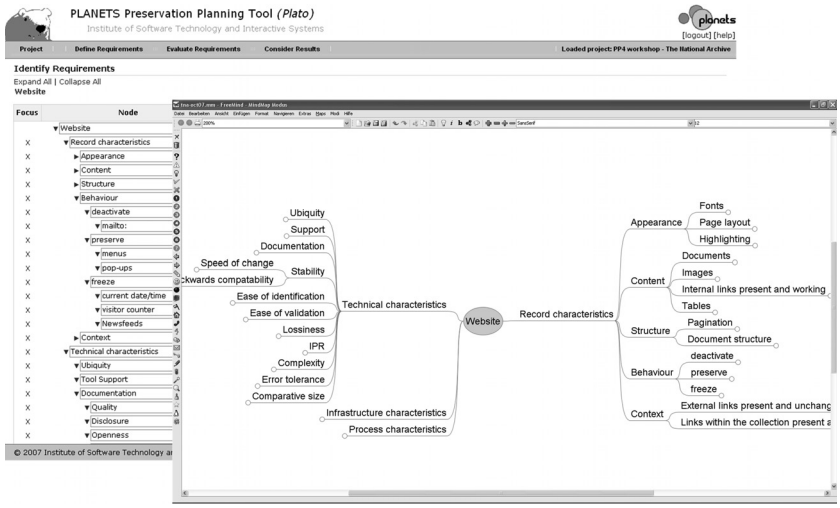


Fig. 2. Requirements definition in Plato

supports loose coupling of services and registries through standard interfaces and provides common services such as user management, security, and a common workspace. Based on this technical foundation, the aim is to create an interactive and highly supportive software environment that advances the insight of preservation planners and enables proactive preservation planning.

Figure 1 illustrates the preservation planning environment, putting the described workflow in the working context of services and registries as they are currently being implemented. It shows three main aspects: (1) Integrating registries for information discovery; (2) Integrating services for preservation action and characterisation of objects; and (3) Proactively supporting the planning with a knowledge base that holds reusable patterns and templates for requirements recurring in different planning situations.

The right choice of samples that are representative for the collection under consideration is essential, as any skewed representation might lead to wrong results. **Collection profiling services** based on characterisation services and format registries can inform the selection process and ensure the right stratification of samples. **Risk assessment services** can further assist by quantifying both the inherent risks of object formats and the salient risks present in the objects which are of particular relevance to a specific file format, such as the number of pages for some document formats or the presence of transparency layers in images.

The specification of requirements in a tree structure is often done in a workshop setting. This is supported by both a flexible web interface as depicted in Figure 2 and a direct tree import from mind-mapping software[5]. The knowledge

[5] `http://freemind.sourceforge.net`

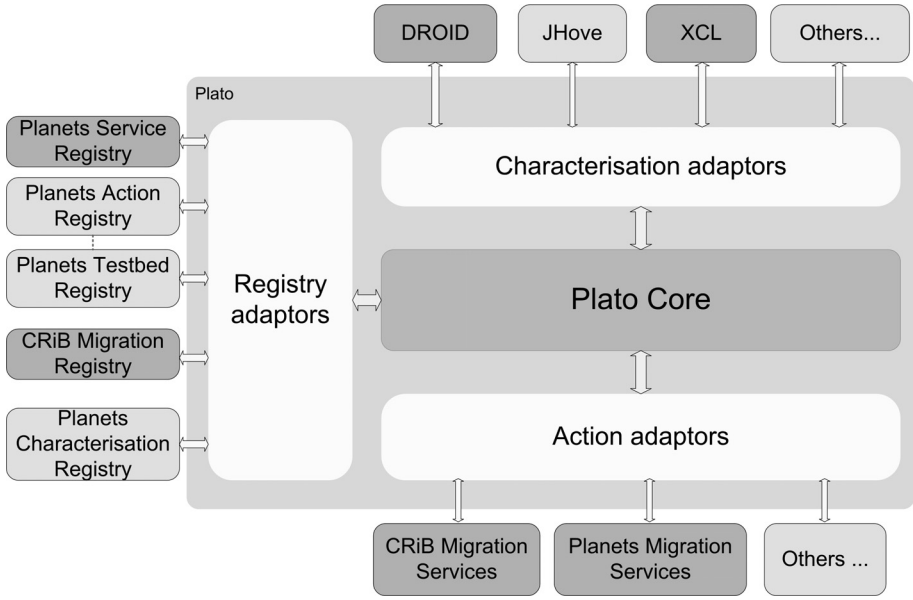


Fig. 3. Overall integration architecture

base provides recurring fragments and templates, such as process requirements for an archival institution or essential object characteristics for electronic documents in a library, to assist in the process of tree creation.

5 A pluggable evaluation framework

5.1 Introduction

While the preservation planning approach and the supporting planning tool outlined above provide considerable support and guidance, we need a link to existing tools and services performing preservation action and characterisation as well as a dynamic integration of information from different, partly heterogeneous information sources (registries). This section outlines an integration architecture and a pluggable framework for automating the evaluation of preservation actions in the described context.

Figure 3 shows the overall building blocks of the architecture. Three types of adaptor layers are needed:

1. **Registry adaptors** provide access to information sources. This primarily refers to registries holding information about preservation action tools and services, but also includes access to preservation characterisation registries that hold information such as risks of file formats.

2. **Action adaptors** are needed for accessing (remote) preservation action tools and services that come in different flavours and varying form. A number of migration services are available online that convert objects[3]. The Planets preservation action tool registry will contain extensive metadata and benchmark experiences from conducted preservation experiments.
   On the other hand, emulators can be a viable alternative in certain instances. Remote access to emulation can support the evaluation and the decision whether or not the additional effort for setting up an emulation environment is both feasible and valuable in a given planning situation.
3. Finally, **characterisation adaptors** access tools and services which can identify file formats, assess the risks of digital objects, extract some or all of their properties and compare these.

These characteristics extracted by the above mentioned characterisation tools and services can be of considerable heterogeneity and complexity. Moreover, the tools are just emerging and rapidly evolving. We thus propose a flexible pluggable architecture for the automated evaluation of objectives and criteria leveraging these services. The basic concept is to enable the dynamic attachment of 'plugs' to criteria in the objective tree, where a plug provides an evaluation value for a defined criterion. We identify three types of plugs:

1. **Comparators** are used for comparing significant properties of objects to validate that the application of a preservation action has not led to a breach of authenticity by destroying or changing a significant characteristic of the original object. To this end, they rely on characterisation tools and services and combine the outputs of these to evaluate changes in the resulting object.
2. **Evaluators** extract and analyse information about either an object or a preservation action tool and provide an evaluation value for a specific characteristic. This could be for example a risk assessment of a target file format when doing migration planning.
3. **Modellers** are used for specifying more complex relationships between key influence factors and their impact on outcomes and evaluation results. For example, cost factors can be combined in cost models to produce an estimate of the costs needed to implement a specific preservation strategy.

While comparators and evaluators need to rely on characterisation tools and services, modellers will mostl likely be largely independent of these. However, the key factors that are used within modellers might be subject to a monitoring process such as technology watch, as is outlined in Section 5.4.

5.2 Comparators

Validating the content of objects before and after (or during) a preservation action is one of the key questions in digital preservation.

While different tools are available, the main focus of our work are the characterisation tool JHove and the eXtensible Characterisation Languages [6] developed within the Planets project.

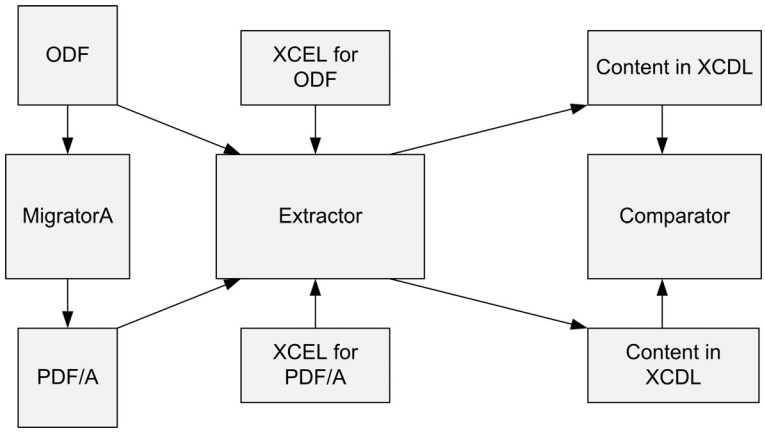


Fig. 4. Using XCL to compare migrated documents

Figure 4, taken from [6], shows a scenario for applying XCL in the context of format migration. After converting a document from ODF to PDF/A, the XCDL documents of the original and the transformed object can be compared using an interpretation software. A comparison tool ('Comparator') for XCDL documents is currently under development. Key objectives are the property-specific definition of metrics and their implementations as algorithms in order to identify degrees of equality between two XCDL documents. In its core functionality the comparator loads two XCDL documents, extracts the property sequences and compares them according to comparison metrics which are defined with respect to the types of the values in the value sets.

To allow the usage of this mechanism within the planning procedure, we need to connect characteristics and comparison metrics to the requirements and criteria defined in the objective tree. The different layers of this conceptual mapping are outlined in Figure 5, which spawns the bridge from objects and their characteristics to overall goals and how they can be broken down to more precise requirements and measurable criteria. The two trees need to be modelled in such a way that they can be connected; furthermore, comparison metrics and mapping structures are necessary to support the quantified and automated evaluation of criteria.

5.3 Evaluators

Evaluators provide characteristics of either objects or actions. A prime example for the first category is risk assessment of objects and object formats. Analysing the characteristics of preservation action services, such as measuring the performance of migration tools or services, falls into the second category.

Risk assessment services are being developed within the Planets project; an exemplary evaluation plug could leverage these services to perform risk assessment on the sample objects and their transformed counterparts. The risk assess-

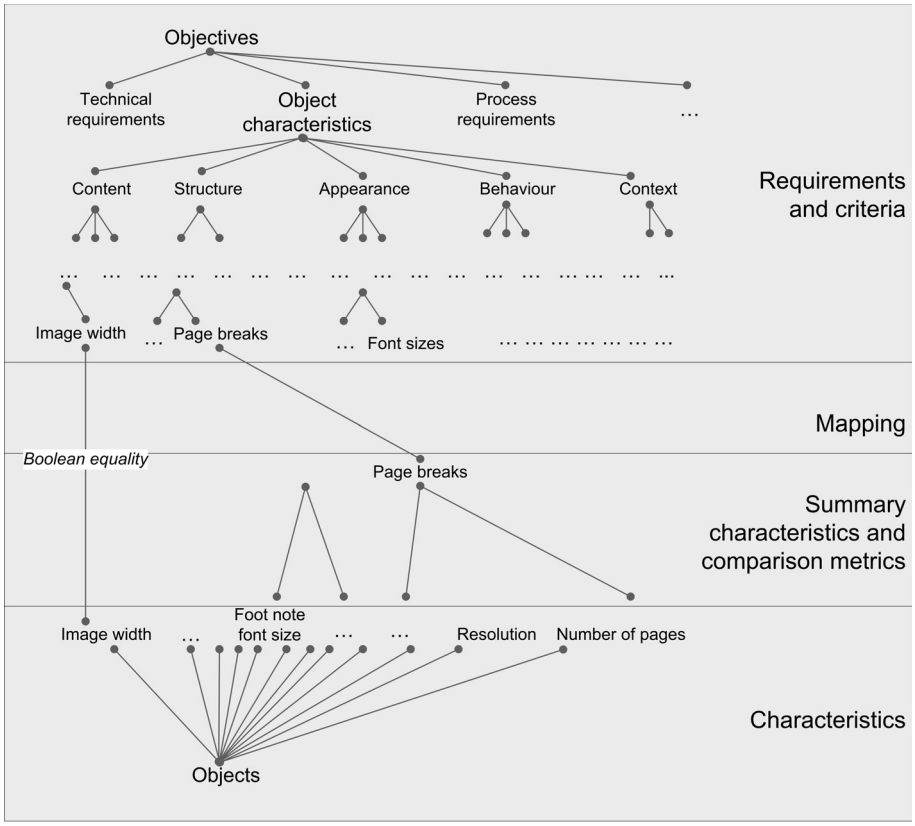


Fig. 5. Connecting object properties to objectives and criteria

ment service in the Planets characterisation framework addresses two categories of risks: (1) General risks of formats, such as complexity or lack of documentation, and (2) Risks that can apply to objects of a certain kind. For example, Word documents with more than 1000 pages may be much more difficult to preserve than short documents.

5.4 Modellers and Watchers

The previous sections have outlined how to analyse, characterise and compare digital objects and actions. This section describes how complex relationships between influence factors can be represented and suggest a mechanism for supporting the assessment of change impact.

Our approach uses **modellers** as the third category of plugs to connect influence factors such as cost factors to criteria in the objective tree. For example, cost factors can be combined through different cost models such as the LIFE and

LIFE2 models[2] to calculate an estimate of the costs of preserving one object for a specified time span.

While evaluators and comparators rely on input from characterisation services, modellers might be used as input to a different category of plugins. **Watchers** extract information from the environment, thus monitoring the environment with respect to specific parameters that influence preferences and decisions. The most prominent example in this context is technology watch, where aspects such as the distribution of file formats are monitored and warnings can be raised when specific thresholds are exceeded.

Watchers are an important basis for continuous monitoring and iterative planning as shown in Figure 1. To this end, thresholds could be defined on various levels that trigger an alert when exceeded. The input parameters to the modeller plugs mentioned above could then be captured, modelled and monitored as well through watchers, leading to a continuous recalculation of the modeller plugs. These in turn can trigger an alert leading to a re-evaluation of preservation solutions and a possible update of the preservation plan.

6 Discussion and Outlook

This paper outlined a flexible and extensible evaluation framework for automating the preservation planning procedure. The concept builds on a solid preservation planning methodology and aims at improving both automation of the workflow and traceability of key influence factors to ensure their impact can be assessed. It consists of an integration architecture and a series of so-called *plugs* which can be attached to criteria for measuring key influence factors that impact preservation planning decisions. We described the main architecture, discussed which issues we deem critical for the successful implementation of this framework, and outlined the next steps in this direction.

The main issues foreseen in completing this work are threefold.

1. The correct trade-off between flexibility and generality on one side, and the specific information needs of algorithms on the other side, can be difficult to find. Similarly, the timely availability of input information needed for the evaluation at the right point in time can be difficult. For example, benchmarking migration services will imply that benchmark results are captured as metadata during service execution; these metadata need to be analysed by the evaluation plug.
2. Related to this, the availability and quality of characterisation services integrated through the evaluation framework is critical to the final quality of the resulting evaluation processes.
3. The mapping between characteristics and requirements needs to bridge the conceptual gap between intellectual properties and technical characteristics.

A thorough conceptual basis is needed to tackle these issues; on the technical level, rapid prototyping can ensure that concepts are validated in an early stage. The successfully implemented architecture should lead to a series of benefits:

– Improved automation of the planning procedure, leading to a considerable reduction in the effort needed to create a preservation plan.
– Improved quantification and understanding of influence factors, leading to a better traceability and improved change impact assessment. This also creates a basis for a continuous monitoring of influence factors through watch functions.
– Researchers and tool developers until now often do not know which kinds of characteristics are significant and need to be extracted and compared automatically. Moreover, they are lacking a common framework of delivering, deploying, testing, distributing, and putting to use algorithms and tools for characterisation, specifically for the comparison of object properties and the characterisation of actions.
  The completed framework can serve as an integration framework for developing advanced services for preservation characterisation and comparison algorithms.
– Analysis of existing and missing services can serve as a gap analysis pointing at problems and thus providing a research agenda for supporting the evaluation of preservation services.

The next steps in developing the described framework are as follows.

– Analyse existing objective trees and their criteria to verify the completeness of the approach with respect to potential influence factors;
– Build prototypical implementations of each type of plug; and
– Build a software infrastructure that supports the dynamic integration of plugs.

Acknowledgements

Part of this work was supported by the European Union in the 6th Framework Program, IST, through the PLANETS project, contract 033789.

References

1. Alves, C., and Castro, J. CRE: A systematic method for COTS components selection. In *XV Brazilian Symposium on Software Engineering (SBES)* (Rio de Janeiro, Brazil, 2001).
2. Ayris, P., Davies, R., McLeod, R., Miao, R., Shenton, H., and Wheatley, P. The life2 final project report. *LIFE Project* (2008). `http://eprints.ucl.ac.uk/11758/`.
3. Becker, C., Ferreira, M., Kraxner, M., Rauber, A., Baptista, A. A., and Ramalho, J. C. Distributed preservation services: Integrating planning and actions. In *Research and Advanced Technology for Digital Libraries. Proceedings of the 12th European Conference on Digital Libraries (ECDL'08)* (Aarhus, Denmark, September 14–19 2008), B. Christensen-Dalsgaard, D. Castelli, B. A. Jurik, and J. Lippincott, Eds., vol. LNCS 5173 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, pp. 25–36.

4. Becker, C., Kolar, G., Kueng, J., and Rauber, A. Preserving interactive multimedia art: A case study in preservation planning. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers. Proceedings of the Tenth Conference on Asian Digital Libraries (ICADL'07)* (Hanoi, Vietnam, December 10-13 2007), vol. 4822/2007 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 257–266.
5. Becker, C., Kulovits, H., Rauber, A., and Hofman, H. Plato: A service oriented decision support system for preservation planning. In *Proceedings of the 8th ACM IEEE Joint Conference on Digital Libraries (JCDL'08)* (2008).
6. Becker, C., Rauber, A., Heydegger, V., Schnasse, J., and Thaller, M. A generic XML language for characterising objects to support digital preservation. In *Proc. 23rd Annual ACM Symposium on Applied Computing (SAC'08)* (Fortaleza, Brazil, March 16-20 2008), vol. 1, ACM, pp. 402–406.
7. Becker, C., Strodl, S., Neumayer, R., Rauber, A., Bettelli, E. N., and Kaiser, M. Long-term preservation of electronic theses and dissertations: A case study in preservation planning. In *Proceedings of the 9th Russian Conference on Digital Libraries (RCDL 2007),* (Pereslavl, Russia, October 2007). `http://rcdl2007.pereslavl.ru/en/program.shtml`.
8. Brody, T., Carr, L., Hey, J. M., Brown, A., and Hitchcock, S. PRONOM-ROAR: Adding format profiles to a repository registry to inform preservation services. *International Journal of Digital Curation 2*, 2 (November 2007), 3–19.
9. Farquhar, A., and Hockx-Yu, H. Planets: Integrated services for digital preservation. *International Journal of Digital Curation 2*, 2 (November 2007), 88–99.
10. Ferreira, M., Baptista, A. A., and Ramalho, J. C. An intelligent decision support system for digital preservation. *International Journal on Digital Libraries 6*, 4 (July 2007), 295–304.
11. Hunter, J., and Choudhury, S. PANIC - an integrated approach to the preservation of complex digital objects using semantic web services. *International Journal on Digital Libraries: Special Issue on Complex Digital Objects 6*, 2 (April 2006), 174–183.
12. ISO. *Open archival information system – Reference model (ISO 14721:2003).* International Standards Organization, 2003.
13. Kontio, J. OTSO: a systematic process for reusable software component selection. Tech. rep., College Park, MD, USA, 1995.
14. Kontio, J. A case study in applying a systematic method for COTS selection. In *Proceedings of ICSE-18* (1996), pp. 201–209.
15. Lawrence, G. W., Kehoe, W. R., Rieger, O. Y., Walters, W. H., and Kenney, A. R. Risk management of digital information: A file format investigation. CLIR Report 93, Council on Library and Information Resources, June 2000.
16. Mellor, P., Wheatley, P., and Sergeant, D. Migration on request, a practical technique for preservation. In *Proceedings of the 6th European Conference on Digital Libraries (ECDL'02)* (2002), M. Agosti and M. Thanos, Eds., Springer, pp. 516–526.
17. Mohamed, A., Ruhe, G., and Eberlein, A. COTS selection: Past, present, and future. In *Proc. ECBS '07* (2007), pp. 103–114.
18. Ncube, C., and Maiden, N. A. M. PORE: Procurement-oriented requirements engineering method for the component-based systems engineering development paradigm. In *Development Paradigm. International Workshop on Component-Based Software Engineering* (1999), pp. 1–12.
19. Rolland, C. Requirements engineering for COTS based systems. *Information and Software Technology 41* (1999), 985–990.

20. Rothenberg, J. *Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation.* Council on Library and Information Resources, January 1999. http://www.clir.org/pubs/reports/rothenberg/contents.html.
21. Rothenberg, J., and Bikson, T. Carrying authentic, understandable and usable digital records through time. Tech. rep., Report to the Dutch National Archives and Ministry of the Interior, The Hague, Netherlands, 1999.
22. Saaty, T. L. How to make a decision: the analytic hierarchy process. *European journal of operational research 48*, 1 (1990), 9–26.
23. Strodl, S., Becker, C., Neumayer, R., and Rauber, A. How to choose a digital preservation strategy: Evaluating a preservation planning procedure. In *Proceedings of the 7th ACM IEEE Joint Conference on Digital Libraries (JCDL'07)* (June 2007), pp. 29–38.
24. Strodl, S., and Rauber, A. Preservation planning in the OAIS model. In *International Conference on the Digital Preservation (IPRES 2007)* (Beijing, China, October 2007).
25. Testbed, D. P. Migration: Context and current status. White paper, National Archives and Ministry of the Interior and Kingdom Relations, 2001.
26. The Center for Research Libraries (CRL), and Online Computer Library Center, Inc.(OCLC). Trustworthy Repositories Audit & Certification: Criteria and Checklist (TRAC). Tech. Rep. 1.0, CRL and OCLC, February 2007.
27. UNESCO. UNESCO charter on the preservation of digital heritage. Adopted at the 32nd session of the General Conference of UNESCO, October 17, 2003. http://portal.unesco.org/ci/en/files/13367/10700115911Charter_en.pdf/Charter_en.pdf.
28. van der Hoeven, J., and van Wijngaarden, H. Modular emulation as a long-term preservation strategy for digital objects. In *5th International Web Archiving Workshop (IWAW05)* (2005).
29. van Lamsweerde, A. Goal-oriented requirements engineering: A guided tour. In *Proceedings RE'01, 5th IEEE International Symposium on Requirements Engineering* (Toronto, Canada, 2001), pp. 249–263.

Preserving Interactive Content: Strategies, Significant Properties and Automatic Testing

Mark Guttenbrunner

Vienna University of Technology, Vienna, Austria
http://www.ifs.tuwien.ac.at/dp
guttenbrunner@ifs.tuwien.ac.at

Abstract. Preserving digital data is becoming a more important issue with the growing amount of data that only exists in digital form. But not only static documents that can have representations in paper or analog tape have to be kept accessible, also interactive forms of data have to be preserved. Interactive fiction, also known as video games, and digital art are just two forms of interactive content that have to be kept alive for historic reasons. Business and scientific applications have to be preserved as well to have data in databases or in documents which can't be altered for authenticity reasons accessible over a long term. But also documents can contain input fields and interaction ranging from simple formulas to full-fledged interactive 3D-animations and beyond. Emulation is one of the strategies to keep applications for old hardware usable on modern systems. As it is not the only strategy to deal with the digital preservation of complex content, we have to be able to evaluate the different preservation strategies.
This paper identifies significant properties of the different types of interactive objects that are analyzed in order to determine optimal preservation solutions. Then various migration and emulation strategies for preserving complex objects over a long term are shown. We discuss methods to automatically test alternatives also by utilizing characterization languages. Then we outline what further steps have to be taken to verify the significant properties and define features that have to be extractable from environments to support automatic testing.

1 Introduction

Digital preservation is a pressing issue for all kinds of documents. But not only static documents have to be considered, complex interactive content has to be preserved for future use as well. Interactive content comes in various forms, some of which are obvious like video games or interactive art. But also application software, dynamic documents and web-pages present some form of interactive content.

For evaluating the suitability of a digital preservation alternative for complex interactive objects the significant properties of an object have to be identified

and analyzed first. The significant properties of static documents usually differ from those of dynamic and interactive content. While the appearance of the first makes it often possible to migrate the contents to other formats, the task is more complex for interactive content. Potential loss has to be investigated very closely, as for example loss of interaction can render a digital art object completely useless. Visual and audible properties as well as interaction with the object have to be preserved. While emulation might look as an obvious choice, other strategies have to be considered as well. Even with the same significant properties for different types of complex content the weighting of importance of these properties for preservation can be different depending on the type and the designated user community. To support automatic testing of alternatives in the process of preservation planning the significant properties have to be stored in a standardized form.

This article is structured as follows. First an overview of related work is given in Section 2. Then we categorize the different types of interactive content and discuss the challenges that are associated with them in Section 3. A discussion on significant properties of the various categories is done in Section 4. In Section 5 we discuss some possible strategies to preserve the identified properties. Section 6 presents some ideas on the automatic testing of alternatives and finally in Section 7 we discuss the results and give an outlook on future work to be done.

2 Related Work

Most digital preservation projects in the past concentrated on the preservation of static documents. One approach to preserve complex multimedia art was done by the Guggenheim museum with the *Variable Media Initiative*. One outcome was the *Variable Media Questionnaire*, a questionnaire for artists and collectors of art which included descriptive elements needed for recreating the artwork. The research concluded in the *Variable Media Network*[1]. The variable media paradigm lets the artists choose between different strategies for preserving their art. The available options are storage, emulation, migration and reinterpretation. Hunter et. al. describe in [6] an approach to use a combination of emulation or migration and the use of metadata for describing the digital object. With the PANIC[2] project a prototype of a web service based digital preservation tool for semi-automatic preservation of complex multimedia objects is presented in [7]. The PANIC project concentrates on objects composed from different content and does not focus on interactive objects.

The term *Emulation* refers to the capability of a device or software to replicate the behavior of a different device or software. It is possible to use hardware to emulate hardware or to use software to emulate software. In this article the meaning of *Emulator* is used as defined in [11] for a program that virtually recreates a different system then the one it is running on.

[1] http://variablemedia.net/
[2] http://www.metadata.net/panic/

44

The concept of using emulation for digital preservation is to keep the data in its original, unaltered form and keep using the software originally used to display the data. This software has to be run on the operating system and the operating system on the hardware it was developed for. To keep this chain alive, an emulator for the original hardware is produced. Methods to establish emulation as a long term strategy for digital preservation using chaining, rehosting and Virtual Machines are discussed in [11].

An approach for a Universal Virtual Machine is the *Universal Virtual Computer (UVC)* by IBM ([5]). It introduces a virtual machine which is simple enough that it can be easily implemented but still sufficient for preserving digital data. The digital data is stored together with a program that can be run on the UVC but that is written when the data is archived. On a future system where the data will be restored an implementation of the UVC is hosted and the program which was stored with the data is used to extract the data from the record. A proof of concept for this approach has been done on the archiving of PDF-documents.

Emulation can take place on different levels (software, operating system or hardware) ([10]). The most accurate approach to the emulation of console video games is most probably the emulation on a hardware level. Especially video games released later in the life cycle of a video game console are using system specific features on a very low hardware level to produce the best results with the then very well known system behavior. Video game consoles do not differ in the used hardware components like personal computers, so programmers can use very time restrained code and optimize the results to the specific system. Emulating a video game console on a different level than hardware would result in low compatibility for the game software. This approach uses software to reproduce the characteristics of hardware components and is not to be confused with emulating hardware using different hardware, which would not solve the digital preservation problems outlined before.

An approach to developing an emulator on a hardware level is discussed as a conceptual model in [13] as *modular emulation*. It suggests the use of a modular emulator which uses a component library and an emulator configuration document to bind the different components for a specific system to an emulated system. A component for which an emulator once has been written can be used for other systems using the same component. The ideal case is to write a new emulator by creating a configuration file and using all the existing components. The modular emulator is run on a Universal Virtual Machine (UVM) as suggested before. A controller program is used to start the UVM and loads the required components and the emulator.

An emulator which uses the modular emulation approach named *Dioscuri*[3] is currently under development. It is currently able to emulate a machine based on the Intel 80x86 processor with basic input/output facilities. Dioscuri is written in Java and runs on the Java Virtual Machine. The goal of the project is a version

[3] http://dioscuri.sourceforge.net/

45

of Dioscuri that is able to replace a reference workstation running Microsoft Windows 2000 ([9]).

A practical experiment on how to use emulation to recreate interactive art is presented in [8]. The original piece of art called *The Erl King* (1982-85) by Grahame Weinbren and Roberta Friedman consisted of obsolete and generic hardware and software. It presented itself as an ideal candidate for an emulation project as the original software was written by the artist, so it was a very high priority to preserve the original code.

Becker et. al. present in [2] case studies on sample objects of interactive multimedia art from the collection of the Ars Electronica[4]. The PLANETS preservation planning approach is used for evaluating applicable digital preservation strategies for interactive multimedia objects. In [4] a case study on preserving console video games was done to evaluate existing emulators for their suitability as digital preservation alternatives. Both case studies identify significant properties of complex content.

A generic language for characterizing objects and describing their properties can be found in [3] as *XCDL* (extensible characterization definition language). Another language developed by IBM is called DFDL (data format description language) [1].

3 Types of Interactive Content

This section describes four types of interactive content. The boundaries are not fixed, as it is sometimes difficult to decide what category an object belongs to (e.g. video games vs. games produced as digital art, application vs. interactive document which themselves can be seen as applications again).

3.1 Application Software

Preserving application software can be necessary for various reasons: Using original documents in their unaltered form for authenticity, running scientific software to reproduce experimental results, historical reasons, access to data in databases. Most of these reasons require interactivity. Another issue are applications distributed over the network, as all components and the interaction between them have to be preserved.

3.2 Dynamic Documents

Documents not only contain data, but logic in some form as well. This can range from simple format qualifiers to complex programs with input-fields for users and calculated graphs. Examples of dynamic documents include web pages using Java-script, spreadsheet documents using formulas, PDF-documents with interactive 3D Animations. For some applications it might be possible to convert the document to a static format comparable to a print-out while for others the interactive elements have to be preserved.

[4] http://www.aec.at

46

3.3 Video Games

From simple games played on early gaming hardware to virtual on-line worlds the requirements to preserve video games differ essentially. Legal issues, unavailable source code and proprietary user interface hardware are only some of the challenges when dealing with interactive fiction. Preserving the non-technical requirements like the feel-aspect or the environment of a playing-experience (e.g. arcade-games in a bar) have to be considered as well. They are a big part of experiencing a game the way it was supposed to be. A great variety of platforms (PC, game consoles, arcade games, mobile telephones, on-line-games) have to be considered when trying to preserve video games.

3.4 Interactive Digital Art

Digital art can contain interactive components just like video games. Additionally it can be implemented on proprietary hardware or even have hardware as part of the art. Preserving interactive digital art can normally only be done on a per piece basis. Art is usually done in a specific context, so this has to be kept in mind when trying to preserve art as well.

4 Significant Properties of Interactive Content

To evaluate a digital preservation alternative it is necessary to know the properties of an object which are significant and which have to be preserved. These properties can be technical as well as social properties. Depending on the type of object and the designated use the weighting of the importance of meeting specific requirements can be different.

Significant properties of all kinds of software include visual and audible properties. All kinds of interactive input possibilities have to be considered. In case of application software and dynamic documents these are e.g. form fields, icons, menus and mouse and keyboard for input. For video games and digital art this can be menus, icons on the user interface, the response and support of hardware like gaming hardware, video cameras, sensors, motion detectors and mouse and keyboard again.

Functionality is an important part of software preservation. In case of applications or documents it means accessing the data while with video games and digital art the playing experience, response to input and audible/visual characteristics are important.

4.1 Application Software

For application software there usually is more weighting on the functionality than on audible or visible characteristics. The original look and feel is less important then being able to access the data. One exception to this rule is the preservation of application software for historical reasons.

47

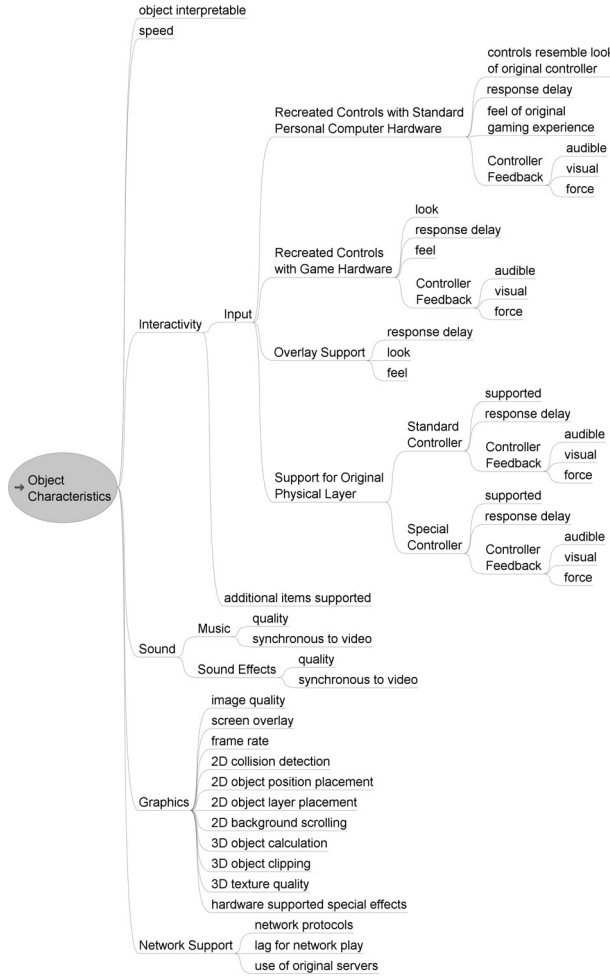


Fig. 1. Significant object properties for console video games [4]

4.2 Dynamic Documents

In dynamic documents the contents and appearance can be important properties with functionality on the original data as well. A good example would be a spreadsheet with a graph thats calculated by using data in the spreadsheet taking current date into account. Of course it also depends on the application of the spreadsheet, if the functionality should always use the current date or if the date should be frozen on ingest reflecting data at that point in time.

4.3 Video Games

Besides visual and audible characteristics the original speed of the game and response to input is very important to re-create the original feel of video games.

48

But also social aspects have to be considered. Playing a game feels a lot different using keyboard and LCD-screen in the office than playing it in a smoke-filled bar on an arcade machine with controls where one can forcefully push buttons. Another thing to be considered for video games is the fact that the gaming experience not necessarily relies on the original graphics and sounds. Updated versions of games for modern hardware might still bring back the original feeling of playing a game without the feel that the game looks dated. A subtree of significant object properties for console video games as developed in a case study in [4] is shown in Figure 1

4.4 Digital Art

For digital art visual and audible characteristics are important. If a piece of art uses analog and digital material and sensors, then the synchronous interaction between this parts has to be preserved. Methods of interaction with the object have to be preserved as well.

5 Strategies

The two main strategies of digital preservation are migration and emulation. With complex content various forms of these strategies are possible:

Source Ports Migrating the digital object to a different system by re-compiling the source code on that platform. This preservation action has to be performed for every object and can only be done when the source code is available. Complexity and effort are very high. This is a potential strategy for all kinds of complex content.

Simulation If source code is not available software can be reimplemented either by using design documents or by re-engineering the original behavior. Like source ports this is a time consuming alternative and a difficult task for complex software. One possible application of this strategy is interactive digital art.

Video Approach Migrating the visual and audible characteristics to video by filming a digital object. While all the interactivity to the object is lost, this strategy might gives a very good idea about the digital object. As it is very inexpensive it is a good alternative for applications with no importance on interaction to the object.

Database Preservation If the interactivity of software does not have to be preserved and data and application logic are strictly separated, the database content can be stored in a standardized format over a long term for either simple querying or re-import to a database.

Low-Level Emulation Emulating a system on a hardware level makes all applications for this system usable by developing only one piece of software. This strategy preserves all the interactivity. It can be a long-term strategy by making sure the emulator can be preserved as well for example by developing it for an emulation virtual machine. A big disadvantage of emulation is the required knowledge about using the original system.

6 Testing

With various possible strategies for the preservation of interactive content it is necessary to evaluate which alternative should be used for a specific digital preservation situation. One way to compare alternatives is the Planets preservation planning approach described in [12]. It uses requirements trees to identify how well significant properties are met by a specific alternative.

To allow tools that automate parts of the preservation planning process to determine these figures, characterization languages can be used. While tools are able to examine migrated documents, this is more difficult for interactive objects.

Below are two methods to test interactive environments:

- The original visual and audible features are recorded in a digital video format. Interaction (e.g. mouse movement and clicks) are recorded and replayed using a preservation alternative. If no random elements appear (like e.g. in video games) the resulting visual and audible features can be compared. Frame count differences and image resolution can easily be extracted.
- Another way to test environments is having the environment extract features. By defining the significant properties and identifying the properties that can be extracted, this feature could be implemented in e.g. emulators. Possible properties are CPU-cycles per second, frames per second, resolution, information about recorded interaction. To simulate the interaction of a user a recorded user input macro has to be provided and executed by the tool.

7 Discussion and Outlook

In this article the types of interactive complex content were described along with challenges for their preservation for a long term. We defined some significant properties and showed the difference between the types. It showed that common significant properties do exist and provided possible strategies for preserving them.

Describing the significant properties in a generic language allows the use of comparator tools to automatically compare different alternatives. While most technical significant properties can probably extracted either from the tools used for emulation or from running the migrated software on a system, social significant properties have to be manually evaluated. By defining technical properties now, support for automatic extraction can be implemented.

Future work in refining and verifying the significant properties in case studies with a number of alternatives has to be done. These properties then have to be represented in one or more languages for characterization. For this purpose it could be necessary to extend an existing characterization language. Tools for automatic testing of emulators have to be written. Features that have to be extracted from emulation environments have to be defined, so that emulators can provide support for extraction of those to aid automatic comparison.

50

Acknowledgements

Part of this work was supported by the European Union in the 6th Framework Program, IST, through the PLANETS project, contract 033789.

References

1. Beardsmore, A. Schema description for arbitrary data formats with the data format description language. *Enterprise Interoperability II* (2007), 829–840.
2. Becker, C., Kolar, G., Küng, J., and Rauber, A. Preserving interactive multimedia art: A case study in preservation planning. In *Proceedings of the Tenth International Conference on Asian Digital Libraries* (2007), p. (accepted for publication).
3. Becker, C., Rauber, A., Heydegger, V., Schnasse, J., and Thaller, M. A generic XML language for characterising objects to support digital preservation. In *Proc. 23rd Annual ACM Symposium on Applied Computing (SAC'08)* (Fortaleza, Brazil, March 16-20 2008), vol. 1, ACM, pp. 402–406.
4. Guttenbrunner, M., Becker, C., and Rauber, A. Evaluating strategies for the preservation of console video games. In *Proceedings of the Fifth international Conference on Preservation of Digital Objects (iPRES 2008)* (London, UK, September 2008).
5. Hoeven, J., Van Der Diessen, R., and Van En Meer, K. Development of a universal virtual computer (UVC) for long-term preservation of digital objects. *Journal of Information Science Vol. 31 (3)* (2005), 196–208.
6. Hunter, J., and Choudhury, S. Implementing preservation strategies for complex multimedia objects. In *The Seventh European Conference on Research and Advanced Technology for Digital Libraries* (2003), pp. 473–486.
7. Hunter, J., and Choudhury, S. PANIC: an integrated approach to the preservation of composite digital objects using semantic web services. *International Journal on Digital Libraries 6*, 2 (April 2006), 174–183.
8. Jones, C. Seeing double: Emulation in theory and practice. the erl king case study. *Electronic Media Group, Annual Meeting of the American Institute for Conservation of Historic and Artistic Works* (Variable Media Network, Solomon R.Guggenheim Museum, 2004).
9. KB nationaal archief. Dioscure - digital preservation. Online in Internet, 2007. `http://dioscuri.sourceforge.net/preservation.html` (accessed at September 24, 2008).
10. Rothenberg, J. *Using Emulation to Preserve Digital Documents*. Koninklijke Bibliotheek, 2000.
11. Slats, J. Emulation: Context and current status. Online in Internet, 2003. `http://www.digitaleduurzaamheid.nl/bibliotheek/docs/white_paper_emulatie_EN.pdf` (accessed at September 24, 2008).
12. Strodl, S., Becker, C., Neumayer, R., and Rauber, A. How to choose a digital preservation strategy: Evaluating a preservation planning procedure. In *Proceedings of the 7th ACM IEEE Joint Conference on Digital Libraries (JCDL'07)* (June 2007), pp. 29–38.
13. van der Hoeven, J., and van Wijngaarden, H. Modular emulation as a long-term preservation strategy for digital objects. In *5th International Web Archiving Workshop (IWAW05)* (2005).

Enhancing Music Maps

Jakob Frank

Vienna University of Technology, Vienna, Austria
http://www.ifs.tuwien.ac.at/mir
frank@ifs.tuwien.ac.at

Abstract. Private as well as commercial music collections keep growing and growing. The increasing number of songs in these repositories pose serious challenges to users. PlaySOM and PocketSOM provide map-based access to large audio collections. They provide a quick overview of the whole collection as well as an in-depth view on specific music styles. Furthermore they support the user while exploring and navigating through the collection and provide quick and intuitive playlist creation. But yet, Music Maps have not revealed their full strength. There are still several issues to be solved, such as the continuing growth of collection or multi-user playlist generation. Questions related to these and other issues will be identified and outlined in this paper.

1 Introduction

The immersive grow of private as well as commercial collection of digital audio files has reached a limit where ordinary meta-data based search and browse is no longer sufficient. Several thousand songs can nowadays be stored on personal computers but also on moblie devices, not to speak of the huge amount of music available on commercial audio portals such as iTunes. This huge amount and variety of music calls for novel approaches for searching, browsing and selecting music. Most recent approaches that go beyond textual search and retrieval rely on user-created data such as tags or require social network data. Both techniques suffer from several weaknesses such as the "cold-start" problem that arises for new files in the system.

A novel approach is the usage of Music Maps which arrange and present music on a map-like interface. Based on sophisticated content analysis techniques Music Maps visualise similarities between audio files. This helps to get an overview over large audio collections and provides intuitive and interactive access to them. This novel approach is promising but yet does not reveal its full strength. There are several issues yet to address such as multi user scenarios and the continuing growth of collections.

The remainder of this paper is structured as follows: Section 2 introduces the technical background required for Music Maps. Section 3 will then present the map-based access to large audio collections while Section 4 shows novel applications and interesting issues yet to solve.

2 Technical Background

Music Maps rely on several calculation methods before they yield in intuitive and easy to use interfaces to large audio collections. The creation is basically divided into two steps, both described in the following two sections. In Section 2.3 an experimental approach to bring these techniques to the end user is presented.

2.1 Analysing Music

The first step to Music Maps is the analysis of the audio collection. Different feature extraction methods can be applied to extract meaningfull descriptive data from the audio stream, extracting semantic features from music. These features are then useful for a number of music retrieval applications. Typically, features like loudness, rhythm and timbre (among many others) are extracted by computing the power spectrum of the audio signal to obtain a semantic description of the music content. With these descriptors, classification of music into categories is possible, and also automatic organisation of music collections by similarity (see next subsection). By computing distances between the features of the musical pieces, relations of their acoustic similarity can be derived. Songs having a smaller distance in feature space are highly similar regarding the acoustic and musical aspects described by the features. Thus, with audio features extracted from music a direct retrieval of songs sounding similar to given ones is possible without the need of any manually added meta-data. Moreover, this can be used to automatically generate playlists or help users to explore music libraries more intuitively.

PlaySOM and PocketSOM make use of such an audio feature extractor to create a Music Map. To be more precise a feature extractor extracting Rhythm Patterns and Statistical Spectrum Descriptors is used [2]. These extractors include frequency transformation and psycho-acoustical models, and analyse critical bands and modulation frequencies in order to derive fluctuations and statistical descriptions of frequency bands which the human auditory system is most sensitive to. A Rhythm Pattern comprises modulation strength per modulation frequency (in a range of 0 to 10 Hz) for 24 critical bands. High values for a particular modulation frequency in a number of adjacent bands indicate a specific rhythm in a piece of music. Statistical Spectrum Descriptors are derived by computing several statistical measures from a Bark-scale Sonogram [2]. The resulting features convey information about loudness and timbre and are stored in a feature vector, which is subsequently processed by an algorithm which creates music maps (c.f. Section 2.2). PlaySOM and PocketSOM, however, are not limited to these feature sets and can be extended to use other audio descriptors as well.

2.2 Organizing Music

In order to create a Music Map from the features extracted in the previous step a Self-Organising Map (SOM) is used, organising the music on a rectangular area

in such way that music that sounds similar is located together. A SOM is an unsupervised learning algorithm that is used to project high dimensional data points on a 2-dimensional map [1]. The high dimensional data used as input are the feature vectors extracted from the music signal, as described in Section 2.1.

After analysing the audio files the respective feature vectors are provided to the SOM learning algorithm, which iteratively organises the music on a two-dimensional grid in such a way that similar sounding pieces are grouped close to each other. The algorithm works as follows: The map consists of a definable number of units, which are arranged on a two-dimensional grid. Each of the units is assigned a randomly initialised model vector that has the same dimensionality as the feature vectors. In each learning step a randomly selected feature vector is matched with the closest model vector (winner). An adaptation of the model vector is performed by moving the model vector closer to the feature vector. The neighbours of the winner are adapted as well, yet to a lesser degree than the model vector of the winning unit. This enables a spatial arrangement of the feature vectors such that alike vectors are mapped onto regions close to each other in the grid of the units. Once the learning phase is completed, the feature vector of each music file is mapped to its best-matching unit on the map. By that, similar sounding music is located together, with smooth transitions to other musical styles or genres. Note that the axes of the map have no specific meaning, rather they convey the distances among the music files to each other.

2.3 Web Services

One of the biggest difficulties in Music Information Retrieval is to transfer research results such as feature extraction algorithms from research prototypes to user-friendly and understandable applications. One possible way to tackle this challenge is to use the advantages of the ubiquity of the Internet and provide a web service. Web services are a fine possibility to share feature extraction software easily without giving the details on the implementation out of hands. Furthermore, web services can be integrated into almost every application despite of differences in programming language or execution platform.

Another point is that web services allow to delegate intensive calculations to remote servers, without needing much own resources. Especially on mobile devices, where computational power is still the limiting factor, applications that may otherwise not even be feasible can strongly benefit from web services.

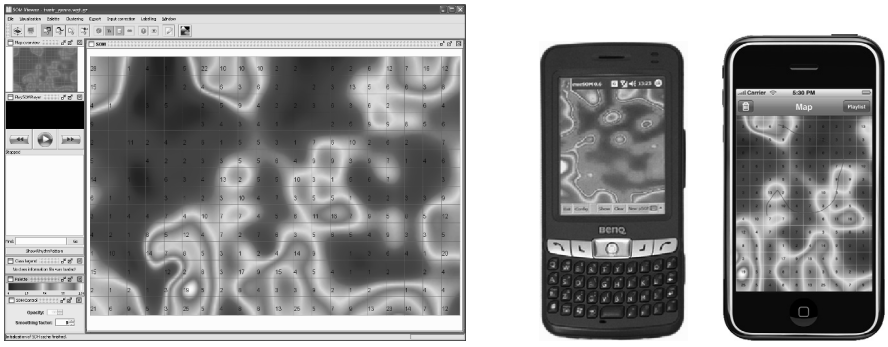
A web service generally consists of two software components: a server providing and a client consuming a specific service. Communication is enabled by the SOAP[1] protocol, which transmits messages in XML format. Our server[2] currently provides two services: feature extraction from audio and the creation of music maps, though adding more services is easily possible. A demo client implementation that can be used to request the service is also provided.

[1] `http://www.w3.org/TR/soap12`

[2] The web service, the demo client and all related documents are available under the following URL: `http://www.ifs.tuwien.ac.at/mir/webservice/`

3 Browsing Music Collections

There are many different ways to browse music collections. The most simple is mere directory based browsing while audio player often provide the feature to browse through different hierarchical structures. This is, however, not the best way to explore a audio collection since it does not show relations between songs that go beyond meta-data matching. Both PlaySOM and PocketSOM address this weakness through displaying the similarity different songs by the distance on the map.



(a) The PlaySOM showing a Music Map (b) PocketSOM on mobile devices

Fig. 1. PlaySOM and PocketSOM

3.1 PlaySOM

The PlaySOM application (see Figure 1(a)) allows users to interact with the Music Map mainly by panning, semantic zooming and selecting of tracks. Users can move across the map, zoom into areas of interest and select songs they want to listen to. It is thus possible to browse collections of a few thousand songs, generating playlists based on track similarity instead of clicking through metadata hierarchies, and listening to those selected playlists. Furthermore it is possible to export them for later use. Users can abstract from albums or genres which often leads to rather monotonous playlists often consisting of complete albums or many songs from one genre. This approach enables users to create playlists based on track not on metadata similarity or manual organisation. By drawing a trajectory on the Music Map it is possible to generate a playlist including smooth transitions between different musical styles. This is especially interesting when browsing very large music collections or when rather long playlists should be generated. Once a user has selected songs and refined the results by manually dropping single songs from the selection, those playlists can be listened to on-the-fly or exported for later use on the desktop machine or even other platforms

56

like PDAs or Multimedia Jukeboxes if the collection is served via a streaming environment. [3]

Furthermore PlaySOM can act as server in conjunction with PocketSOM providing the Music Map as well as the corresponding audio files for streaming. In this case it receives paths, trajectories and playlists sent by the PocketSOM client to display respective replay them.

3.2 PocketSOM

PocketSOM is a viewer application for Music Maps specially developed and adapted for mobile devices and their limited means of interaction. It allows direct interaction with the map using a touchscreen. This gives intuitive access to large audio collections on small devices. [4]

During the evolvement of PocketSOM several different implementations have been created each specially designed for a specific patform. The most recent and sophisticated implementations are ePocketSOM for Windwos Mobile and iSOM for the iPhone/iPod touch (see Figure 1(b)). They are able to load a Music Map over an internet connection from a remote webserver or directly from the PlaySOM application. Furthermore they are able to directly interact with PlaySOM by sending trajectories and paths to be displayed on the map and playlists to replayed central. Finally the above mentioned implementations allow the user full controll of the built-in audio player of the PlaySOM application.

These additional connectivity features allow novel applications which will be outlined in the following Section.

4 Future Work

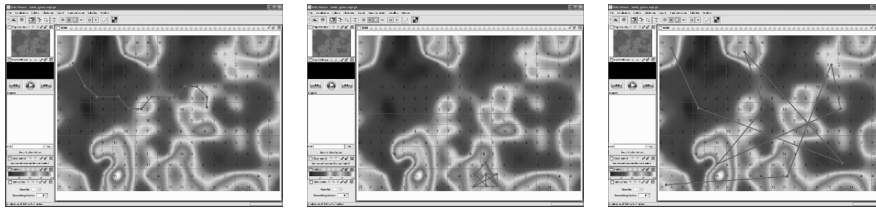
So far, Music Maps on computers and portable devices allow intuitive and interactive access to large music collections. But there are still several issues to solve until Music Maps reveal their full power and benefits.

4.1 Playlist Mapping

The first thing to address is the verification of the path-based playlist generation. The main point is whether user generated "real-world" playlists match the model of trajectories on a Music Map.

So far the assumption is that playlists can be modeled as trajectories on a Music Map. To verify this presumption, user-generated playlists from different sources (e.g. from last.fm[3]) will be visualised on a Music Map containing the songs used in this playlist along with others from the same style. Then the shape of the resulting trajectory will be analysed. So far, the following shapes are imaginable:

[3] `http://last.fm`



(a) Continuous Paths (b) Local Selection (c) Random Jumps

Fig. 2. User-generated playlists mapped on a Music Map.

- **Path:** Playlists do reflect continuous trajectories on the Music Map (Figure 2(a)).
- **local selections:** Playlists stay in a small, isolated area of the Map (Figure 2(b)).
- **random jumps:** Playlists create random long-distance jumps on the Map (Figure 2(c)).
- Any combination of the above mentioned.

Whatever the result of these experiments will be, it will contain valuable information (a) to improve the creation of Music Maps and (b) to understand the human way of perseption of music.

4.2 Expanding Collections

Since audio collections grow constantly, also Music Maps representing them must be constantly adopted. The main problem is that once a user is familiar with "his" Music Map it is very disturbing if the map changes dramatically which might happen when a Music Map is recreated.

As long as only few songs of a similar style already represented on the map are added there is no need to create a new map. Simply adding these songs to the Music Map is sufficient. However, if the range or the distribution of the different styles changes dramatically (e.g. by adding a new musical style) the map has to be retrained. But also in this case the Map should not change completely.

So main questions to address are:

1. At what point does a Music Map need to be recreated? How can this point be automatically determined?
2. How can the system ensure that the map does not change completely?
3. How can the changes on the map be appropriate displayed?

4.3 Path Merging

So far PocketSOM can act as remote control for the PlaySOM application. This is, however, limited to one single user. But when it comes to creating playlists for a group this concept does not reach far enough.

58

To allow multi-user playlist creation the approach is as follows: Multiple users send their trajectories or regions of interests on the map to the central server where these inputs will be further processed. The system tries to merge the received paths to on common playlists that fits all the user's requirements. There are several different different ways to combine paths and points sent by users:

- **Path Concatenation** With this most simple approach paths are concatenated one after the other. This might sound rather unsophisticated but it is, especially in combination with other techniques, challenging to find the best sequence of paths.
- **Path Clustering** With this approach two paths are taken and the average between is calculated and so snapped together. This technique has problems dealing with paths of different length. To avoid such problems paths might be first split into paths of a fixed length and after the clustering reconcatenated.
- **Point Clustering** After converting paths into a series of points these points are then clustered and from the centroids of these clusters a new path is calculated. The main questions for this approach is (a) how many points are used per path, (b) how many clusters are created, and (c) how do the resulting points create a new path?
- **Point Discretisation** Instead of converting paths to their points on the map the grid that lies behind the map is taken into account. Every unit on the grid that is covered by the path is marked. The more ofthen a unit is marked the more weight it will gain in the following clustering process. Again, after calculating the clusters an new path based on the centroids is created. The questions (b) and (c) from the previous point also apply to this approach.

References

1. Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 1995.
2. Thomas Lidy and Andreas Rauber. Evaluation of feature extractors and psychoacoustic transformations for music genre classification. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 34–41, London, UK, September 11-15 2005.
3. Robert Neumayer, Michael Dittenbach, and Andreas Rauber. PlaySOM and PocketSOMPlayer – alternative interfaces to large music collections. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 618–623, London, UK, September 11-15 2005.
4. Robert Neumayer, Jakob Frank, Peter Hlavac, Thomas Lidy, and Andreas Rauber. Bringing mobile based map access to digital audio to the end user. In *Proceedings of the 14th International Conference on Image Analysis and Processing Workshops (ICIAP'07), 1st Workshop on Video and Multimedia Digital Libraries (VMDL'07)*, pages 9–14, Modena, Italy, September 10-13 2007. IEEE.

Session 3

Semantic web middleware

Design and Implementation of Task-based Middleware Execution Engine for JBOWL Text-mining library

Peter Butka, Peter Bednár

Centre for Information Technologies
Technical University of Košice,
Boženy Němcovej 3, 040 01 Košice, Slovakia
{Peter.Butka, Peter.Bednar}@tuke.sk

Abstract. This paper describes design and implementation aspects for extension of our original software system developed in Java for support of information retrieval and text mining with specialized execution engine for different type of tasks. Some of our experiences and specific requirements of the real applications lead us to idea give the system possibility to run the tasks in distributive parallel way. The result of the idea is task-based execution engine, which represents middleware-like transparent layer (mostly for programmers who want to re-use functionality of our package) for running of different tasks in multi-thread environment. The original system is being developed as open source with the intention to provide an easy extensible, modular framework for pre-processing, indexing and further exploration of large text collections. Conceptual architecture of the system is provided as well as details regarding our extension of library like usage of content repository paradigm, representation and encapsulation of tasks, and implementation of the engine itself.

1 Introduction

Our research and education goals in the area of text mining and information retrieval with the emphasis of advanced knowledge technologies for the semantic web resulted in design and implementation of library which is able to efficiently pre-process potentially large collections of text documents with flexible set of available pre-processing techniques, support various types and formats of text (e.g. plain text, HTML or XML), work with text collections in different languages (very different sorts of languages require significantly different approaches in pre-processing phase), support for indexing and retrieval in these text collections (and experiments with various extended retrieval techniques), well-designed interface to knowledge structures such as ontologies, controlled vocabularies or WordNet.

The decision to design and implement a new tool, Java library for support of text mining and retrieval (with acronym JBOWL – Java Bag Of Words Library), was based on the detailed analysis of existing free software tools. More details regarding analysis, design and implementation of original library, as well as several text-mining methods already provided in the system (in time of preparing that paper) can be found

in [1]. We will extend the description with some necessary details in next chapter.

Some of our experiences and specific requirements of the real applications lead us to idea give the system possibility to run the tasks in distributive parallel way. This will give the programmers (application developers) also possibility to re-use library in more complex applications and domains. One example is our need to prepare education portal for text-mining for students on lectures of knowledge discovery from texts in domain of knowledge management. The original library has significant advantages in programming and incorporating of several text-mining methods for most of the problems in this complex domain like classification and clustering of documents, different pre-processing techniques or information extraction methods – this is more individually oriented usage of library, for simple one-problem running of tasks and experiments. On the other hand in our project (project Poznať [2]) the goal is to provide portal for the students (lectures), where many users could run several text-mining tasks with different collections and evaluate the results. This type of application then logically needs to run tasks on machine in multi-thread or distributive way.

The result of this idea for extension is task-based execution engine, which represents middleware-like transparent layer (mostly programmers who want to re-use functionality of our package) for running of different tasks in multi-thread environment. In the next chapter we will introduce conceptual architecture of the system, some comments and ideas regarding our extension of library like usage of content repository paradigm, representation and encapsulation of tasks. Then design and implementation of the engine itself will be provided in more details. At the end of the paper we will provide some future work remarks and conclusions.

2 Conceptual Architecture of JBOWL

JBOWL has the same architecture like standard Java Data Mining API (JSR 73 specification [3]). Also new specification is in preparing phase, but it is not finished yet, so we would stick to JSR73 (with extensions if some new concepts seem to be interesting for our purposes). This architecture has three base components that may be implemented as one executable or in a distributed environment.

- **Application Programming Interface (API)** - The API is set of user-visible classes and interfaces that allow access to services provided by the text mining engine (TME). An application developer using JBOWL requires knowledge only of the API library, not of the other supporting components.
- **Text Mining Engine (TME)** - A TME provides the infrastructure that offers a set of text mining services to its API clients. TME can be implemented as a local library or as a server of client-server architecture.
- **Mining Object Repository (MOR)** - The TME uses a mining object repository which serves to persisting of text mining objects.

TME manages execution of common text mining tasks, e.g. document analyzing, building a model, testing a model, applying a model on new data, computing statistics, and importing and exporting existing mining objects from and to MOR.

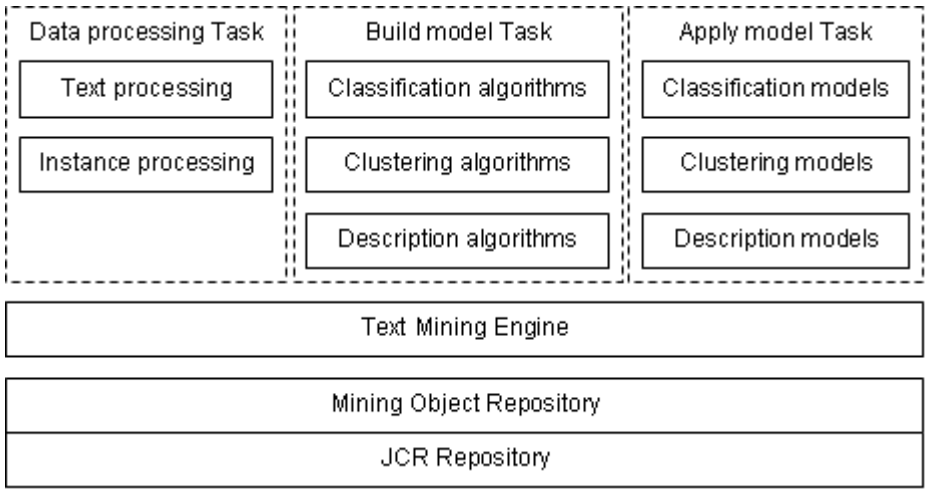


Fig. 1. This is conceptual scheme of JBOWL high-level architecture with three basic components – Text mining tasks (API), Text Mining Engine (TME) and Mining Object Repository (MOR). One of the new aspects in JBOWL architecture is support of JCR (Java Content Repository) in design and implementation of MOR.

Figure 1 follows Data Mining API architecture, but it also shows extension of the system in order to achieve our goals. Main difference to our older version is use of the content repository paradigm in MOR in order to have system more flexible and re-use of data collections and already finished processing of text mining objects clearer and easier. At the bottom of the architecture we can see MOR layer with newly introduced JCR Repository (JCR – Java Content Repository [4]). Next there is TME which will be extended in order to provide multi-thread support. At the top of the conceptual architecture we can see layer of different tasks which represents API layer of the Data Mining API.

2.1 Text Mining Tasks

JBOWL provides set of common java classes and interfaces that enable integration of various pre-processing, classification and clustering methods. The design of the library in case of model-building algorithms (classification and clustering models) distinguishes between algorithms (i.e. SVM, linear perceptron, etc.) and models (i.e. linear classifier, rule based classifier, SOM, etc.). Models are built using algorithms and then created models could be used in applying or evaluation on data inputs. More specific task is transformation of input data – processing of input sets of documents. According to this we have (Fig.1):

- **Data Processing Tasks** – input datasets (sets of textual documents) are processed using these tasks, it means that input documents are recognized, text in documents is tokenized and statistics are computed and saved, vector representation of documents (instances) is prepared according to statistics,

scheme of weighting (tf-idf) is used for weighting of terms, filtering/pruning of terms is available like stop-words filtering, selection of terms, etc.

- **Build Model Tasks** – tasks here are responsible for building of appropriate model using chosen algorithm. Nice feature is that models and algorithms have very common structure and it is possible to re-use them for implementation of new algorithms and their combinations. Several text-mining or pre-processing approaches have been already implemented in our package:
    - Text categorization – several well-know algorithms like SVM, linear classifiers (e.g. perceptron), decision trees and rules induction, kNN, together with ensemble learning methods like boosting and bagging.
    - Clustering methods – kMeans, SOM and related methods (GHSOM), agglomerative clustering
    - NLP methods (especially for pre-processing steps) – ATN networks for deeper text analysis
    - Formal Concept Analysis (FCA) – usage of method itself, additionally hybridized with clustering and description methods
    - Description algorithms – Labelling of SOM models or FCA-based models using LabelSOM, extraction of keywords based on other approaches like analyzing of information gain, etc.
    - Meta-learning approach for text categorization
- **Apply Model Tasks** – models created by algorithms are applied to new data inputs in order to classify new documents, find appropriate cluster to some document, find out contextually similar documents within near formal concepts, use meta-model for automatic choosing of algorithm in different domain, etc. Conceptually, also evaluation of models and their processing (with some evaluation metrics) can be seen in this set of tasks (although in implementation they will be differently modelled).

In order to achieve running of all tasks in multi-thread text-mining engine, it is necessary to model them appropriately, we will show our approach in part related to concrete design and implementation of execution engine in chapter 3.

2.2 Text Mining Engine

As it was already written, TME manages execution of common text mining tasks. It can be done as usage of local library or as running tasks in server-client architecture. Our main point is to extend JBOWL with its own transparent layer for running of tasks in multi-thread and potentially distributive manner, where application developers are able to run tasks easily and they probably do not need to know, where the tasks are really executed, they expect results and place where they can find them.

This is idea almost identical to grid computing paradigm. We have already some experiences with combination of JBOWL library and grid environment (e.g. distributed classification (decision trees) of documents on the Grid [5], parallel/distributed implementation of GHSOM [6] or FCA method [7], etc.), which also implies our decision to logically extend library with multi-thread distributive

support. Our experiences with previously mentioned work have shown some requirements for the application developers like running tasks naturally in multi-thread way, re-use of existing results and datasets in more complex way (every mining object will be connected to node in content repository – see next chapter of the paper for details) and possible extension to prepare engine running on different machine without developer needs to change his code (only setup connection), in order to fulfil middleware-like architecture of application based on text-mining tasks.

2.3 Mining Object Repository

Mining Object Repository (MOR) is used as a persistent storage for the processed text content and all artefacts generated during the text-mining process. Persistent objects include annotations of analyzed texts, data and evaluation statistics, indexed instances, text mining models and task settings. The implementation of the MOR is based on the Java Content Repository (JCR) specifications, which provide seamless integration with the existing content repositories.

The main concept of JCR specification is *Node*, which has associated data *Propertie*s. Text content is stored in the properties of the string type, but JCR also supports other data types like dates, Boolean values, real and integer numbers or arbitrary binary content. Nodes are explicitly organized in the hierarchy, but it is possible to have reference properties to other nodes, which can be used to represent non-hierarchical relations. JCR specification provides strong support for search of the content. Queries can be specified with XPath expressions or in the dialect of the SQL language. Other features of JCR specification, which provide benefits for implementation of the MOR, include support of transactions and type system, which specify constrains for possible properties and sub-nodes.

In order to simplify integration of Jbowl and JCR, MOR contains Mining Object Manager component, which maps Java objects to JCR nodes for serialization/de-serialization. The mapping mechanism is generic and can be used to map arbitrary Java objects to JCR in the similar way, how the object-relational mapping frameworks (like Hibernate) are used to map Java objects to relational database.

3 Design and Implementation of Execution Engine

API (Application Programming Interface) of the Execution Engine is divided to client part and server part to simplify implementation of remote task execution. The main interfaces of the Execution Engine are depicted on the following diagram (Fig.2).

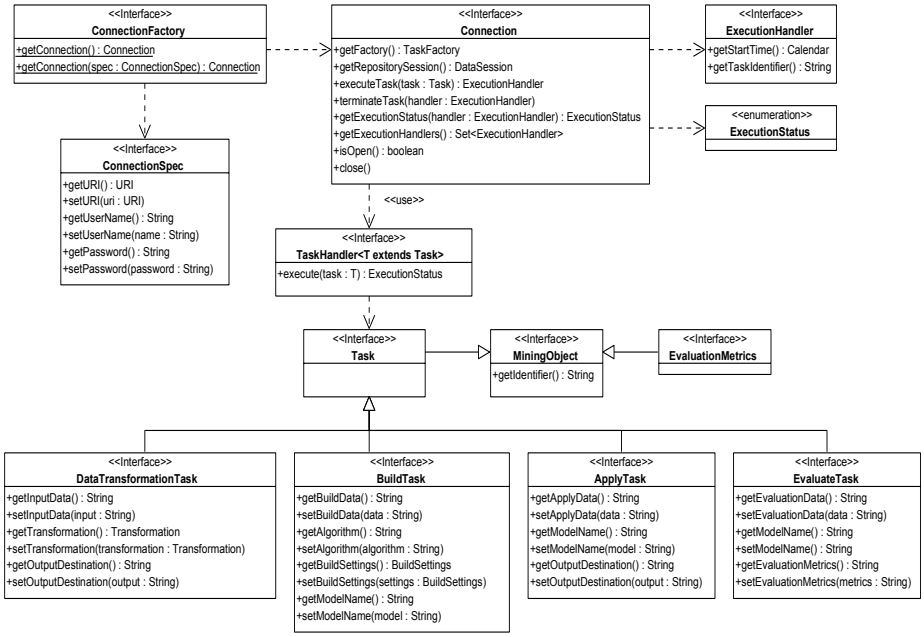


Fig.2. UML design of Execution Engine interfaces.

Connection

To start working with Text Mining Engine (TME), client has to obtain *Connection* object which represent one text-mining session. Connection can be obtained in various ways, for example it can be created directly without user authentication or registered on the client environment using the JNDI. Client can specify details for connection specification like URI of the executed engine in the case that there are more TME instances, user name and password. *Connection* interface will allow client user to

- Obtain Factory class to create new mining objects (i.e. data, tasks, build and task settings etc.)
- Obtain MOR session to save or load mining objects stored in the Mining Object Repository.
- Execute, inspect and terminate text-mining tasks.

Task and TaskHandler

API for tasks is divided to interfaces for task specification (*Task* interface) and for task execution (*TaskHandler* interface). Task objects are part of the client API and follow JavaBeans patterns, which allow simple encoding of the objects in the remote protocols. Task objects specify all parameters required for the specific task, like references to the input data, path where to store output data or models and all associated settings (build settings for algorithms, settings for data processing and settings for evaluation metrics).

Each type of the Task object has associated TaskHandler object, which is responsible to execute this task. According to parameters specified in the Task object, TaskHandler object will create new parallel execution process and perform all operations defined for the task. For example for the Build Model Tasks task handler will load training data, create new instance of the algorithm specified as the task parameter, and pass training data and build settings to the algorithm to produce new text mining model. Model is then stored in the MOR on the path specified as the task parameter.

Execution Handler and Execution Status
When the Execution Engine creates thread for new task and execute task handler, the client invocation of the Execution Engine is immediately finished and task is executed on the background. Execution Engine will return to the client Execution handler, which identify running task and can be used to inspect execution status of the task process or to terminate task.

4 Conclusions

In this paper we have introduced task-based execution engine middleware extension of JBOWL for support of multi-thread/distributed running of text-mining tasks. Important features of the extension is usage of Java Content Repository in Mining Object Repository as basic space for persisting of mining objects like documents and created models. This allows text mining engine layer (Execution Engine) to work with objects in more flexible way and then run (conceptually organized and encapsulated) tasks more easily in parallel/distributed multi-thread way.

More practically, implementation of our Execution Engine as internal and logical component of JBOWL provides support for wide types of applications. In our case engine will be tested in our project, where JBOWL is used as a main text-mining engine behind education portal for students in their study and experimenting with process of knowledge discovery in texts (lectures related to knowledge management technologies).

Acknowledgement

The work presented in the paper is supported by the Slovak Research and Development Agency under the contract No. RPEU-0011-06 and No. APVV-0391-06; by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/4074/07 and No. 1/3135/06; and MVTS project No. Fr/ČR/SR/TUKE07.

References

1. Bednar, P., Butka, P., Paralic, J.: Java Library for Support of Text Mining and Retrieval. In Proceedings of Znalosti 2005, pp.162-169, Stará Lesná, Slovakia (2005)
2. Babic, F., Furdik, K., Paralic, J., Wagner, J.: Podpora procesov tvorby nových znalostí. In Proceeding of DATAKON 2007, pp.198-201, Brno, Czech Republic (2007)
3. JSR 73: Data Mining API: URL: http://www.jcp.org/en/jsr/detail?id=73 (2004)
4. JSR 170: Content Repository for Java API: http://www.jcp.org/en/jsr/detail?id=170 (2005)
5. Janciak, I., Sarnovsky, M., Tjoa, A.Min, Brezany, P.: Distributed classification of textual documents on the Grid. In Proceedings of High Performance Computing and Communications, Second international conference, HPCC 2006, pp.710-718, September 13-15, Munich, Germany (2006)
6. Sarnovsky, M., Butka, P., Safko, V.: Distribuované zhlukovanie textových dokumentov v prostredí Gridu. In: Znalosti 2008, pp.192-203, Bratislava, Slovakia (2008)
7. Butka, P., Sarnovsky, M., Bednar, P.: One Approach to Combination of FCA-based Local Conceptual Models for Text Analysis - Grid-based Approach. In Proceedings of the 6th International Symposium on Applied Machine Intelligence, SAMI 2008, pp.131-135, Herlany, Slovakia (2008)

Semiautomatic workflow composition in grid environment

Tomáš Drenčák

Faculty of Electrical Engineering and Informatics
Technical University of Košice,
Letná 9, 040 01 Košice, Slovakia
tomas@drencak.com

Abstract. This work is based on simple example of Text mining workflow and is initial work for later composition. How can it be fitted into the real life? Consider following situation: User have collection of huge volume data (in our case text documents). He knows what he wants to know about the data (end result) but is not sure about how to achieve this aim even what steps are necessary. In our example we have chosen Text mining workflow where user has a collection of documents and wants to build classification model for later use. He doesn't know which classification model suits best this collection and even what steps are necessary to do (e.g. tokenization etc...).

1 GWorkflowDL

In our implementation we have chosen GWorkflowDL [2] as implementation language. Its name stands for **G**rid **W**orkflow **D**escription **L**anguage. Concept of the GWorkflowDL is based on high level Petri-nets and consists of basically of two building blocks, see Fig. 1:

- places
- transitions

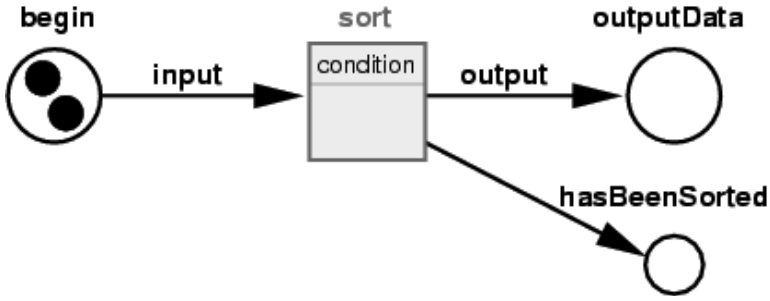


Fig. 1. Example of places connected by transition [1]

Place stands for data and transition are used for transfer of the data between two

places. Typically makes some transformation during the transfer. This transformation is usually a web service call.

2 Workflow refinement

Workflow should be in several different phases, see Fig. 2:

- User request – this is the initial phase of the workflow. User specifies inputs (e.g. collection of documents) and outputs as desired results (e.g. "I want classification model")
- Abstract workflow – in this phase, system refines user request by backward chaining of the web service class operations, e.g. classification model is built by Classification builder service algorithm, Classification builder service requires document matrix etc... which results into → Document matrix service → Classification builder service→ Classification model
- Service candidates – now web service class operations are known. For one web service class operation may exists several services. For example Perceptron, SVM can be service candidates for Classification builder service class operation.

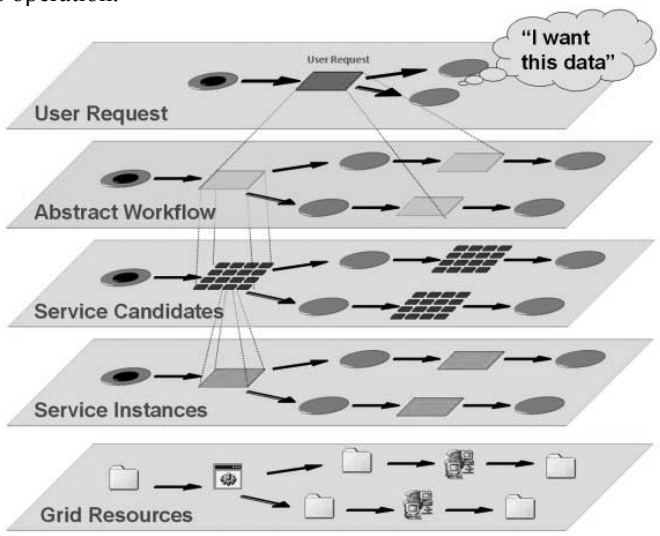


Fig. 2. Phases of the workflow in Workflow refinement

- Service instances – in this phase, service candidates are known. Each service candidate represents set of real web services which can be deployed all over the world, on different quality machines with different quality connection. Some QoS service can be therefore used for choosing the best instances for each service candidate.
- Grid resources – real calls of the web services

Color/Shape	Meaning
Gray	Control elements
Red	Not yet solved
Yellow	Abstract operation
Circle	Place
Rectangle	Transition

Tab. 1. Description of particular elements

2.1 User interaction

As the topic of the work is contains word semiautomatic we expect user to be actively presented on some parts of workflow composition.

Typically there may exist situations when system can't decide on for the right answer by accessible data. In this case user can add some new data to the system or system can propose several choices for the user among which user can choose one, see Fig. 3.

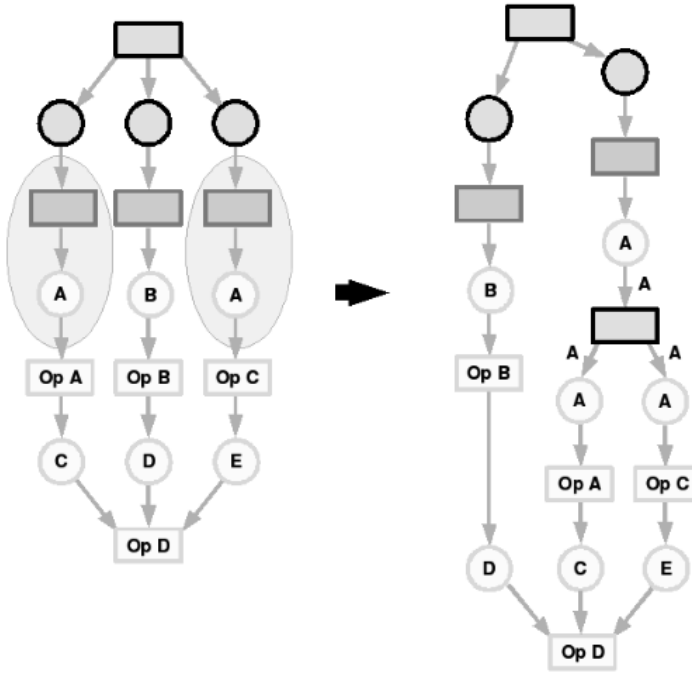


Fig. 3. Workflow refinement in dependency reduction phase. User will be asked to concretize red transitions.

2.2 Implementation

Our implementation consists of several components:

- JBowl text mining library [3]
- SOAP Web services for JBowl (Jbowl-WS)
- Workflow execution environment (Wee)

In this phase of implementation we created SOAP Web services over JBowl library and their corresponding WSDLs. Web services were then connected into executable GWorkflowDL workflow (workflow in service instances phase).

This workflow was then stored in the Wee and executed.

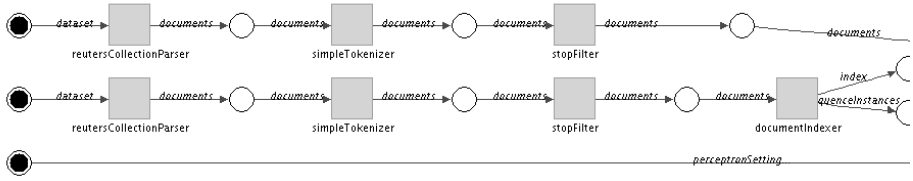


Fig. 4. Text mining workflow - part 1

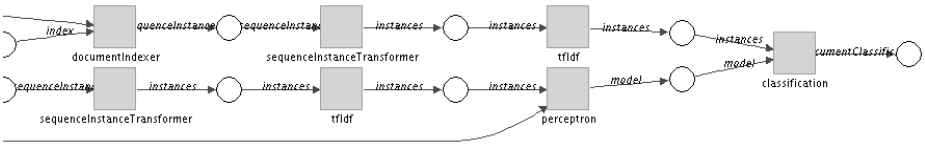


Fig.5. Text mining workflow - part 2

Text mining workflow consists of 2 branches (see Fig. 4 and Fig. 5) – one for model building (train) and one for model evaluation (test). These are connected in the middle by indexing service, which needs to process both train and test documents. Then there are following services:

- Collection parser – parses user input into format which is suitable for the later processing
- Tokenizer – makes tokenization for documents
- Filter – filters tokens in documents. e.g. filters stop words
- Indexer – indexing of the tokens, transforms strings into numbers
- Sequence instance transformer – Indexer transforms strings into sequence of numbers, these are then grouped into groups of "word number" from which we can determine occurrence frequency of the word corresponding to the number. This makes document collection matrix.
- TF-IDF – weighting of the frequencies
- Perceptron – classification builder
- Classification – uses built classification model to classify input documents

3 Conclusions

This work lays grounds for the future improvements to do complete workflow refinement for every phase as specified in section about Workflow refinement.

Acknowledgement

The work presented in the paper is supported by the Slovak Research and Development Agency under the contract No. APVV-0391-06 and by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/4074/07 and No. 1/3135/06; and MVTS project No. Fr/ČR/SR/TUKE07.

References

1. http://www.gridworkflow.org/kwfgrid/gworkflowdl/docs/
2. Martin Alt, Andreas Hoheisel, Hans-Werner Pohl, Sergei Gorlatch: A Grid Workflow Language Using High-Level Petri Nets. In: Proceedings of the PPAM05, Poznan, 2005
3. Bednár P.: JBowl, Java Bag of word library, available at http://sourceforge .net/projects/jbowl/, Accessed: 12th May 2008.

Use of semantic technologies in network embedded middleware

Martin Sarnovsky[1], Peter Kostelnik[1], Jan Hreno[1],

[1] Technical University, Letna 9,040 01 Kosice, Slovak republic
{Martin.Sarnovsky, Peter.Kostelnik, Jan.Hreno}@tuke.sk

Abstract. The paper describes the project FP6 IST Hydra. In general, project aims at development of middleware for intelligent networked embedded system based on service-oriented architecture. The embedded service-oriented architecture will provide interoperable access to data, information and knowledge across heterogeneous platforms. These devices and their local networks will also be interconnected through broadband and/or wireless networks. An implemented Hydra middleware and a toolkit will be validated in real end-user scenarios in three different user domains: Facility management (smart/intelligent homes), Healthcare, and Agriculture. The following paper gives an overview of semantic technologies usage in different areas of Hydra middleware. The main idea of semantics in Hydra is described, as well as overview of concrete mechanisms and models currently being used within the middleware.

Introduction

Hydra ("Networked Embedded System Middleware for Heterogeneous Physical Devices in a Distributed Architecture") is the IST-2005-034891 project funded within the FP6 IST Programme. Hydra project aims at development of middleware for intelligent networked embedded system based on service-oriented architecture, deployable on both new and existing networks of distributed wireless and wired devices [3].

In several aspects the Hydra project builds on the recently emerged idea of the Semantic web – a "web for machines", promising the opportunity for finding and processing information based on employing semantic technologies enabling expression of the semantics of the information. Although the concept of Service Oriented Architectures is not new and has been in use already for several years, the features offered by the concept (e.g. loose coupling, abstraction from the internal design of services, dynamic discovery, platform independence, etc.) represent characteristics the Hydra project can profit from. The promise of Model-driven Architecture is to facilitate the creation of machine-readable models with a goal of long-term flexibility. Since writing platform specific code is replaced by generating the code by transformations, it enables to design models that are independent of the target platform.

Hydra distinguishes two different types of users: Developer users, who will use the Hydra middleware to develop their applications, and end-users, which will use Hydra applications developed by the developer users. In that fashion, we also distinguish two separate application levels: design-time, in which developer users create the applications using SDK (Software Development Kit, which will be also output of the project) and run-time, in which concrete application is running on top of the Hydra middleware.

Semantics in Hydra

Hydra presented the concept of Semantic Devices [1]. The motivation behind the concept is the fact, that the services offered by physical devices are generally designed independently of the particular applications in which the devices might be used. A semantic device on the other hand represents what a particular application would like to have. The basic idea behind such concept is to hide all the underlying complexity of the mapping to, discovery of and access to physical devices. The programmer just uses it as a normal object in his application focusing on solving the application's problems rather then the intrinsic of the physical devices.
Semantics in Hydra are used in both, design-time and run-time. The semantic descriptions of devices and their services are used at design-time to find suitable services for the application that the HYDRA developer is working on – in another words, semantics are used for code generation for semantic devices. In similar fashion, it can be used for code generation for physical devices. The semantic description is used to determine the compilation target. Depending on the available resources of a device, either embedded stubs or skeletons are created for the web service to run on the target device. Semantics in run-time is used mostly for semantic discovery of devices and services. On the other hand, project also studies use of the semantic modeling of security for semantic resolution purposes. The descriptions of semantic devices are based on device ontology.

Device Ontology

Device Ontology is one of the key components in the Hydra middleware. It is equipped with all meta-information and knowledge about devices and device types. Hydra Device Ontology is inspired by the FIPA Device Ontology [4] and initial device taxonomy was extended from AMIGO project [5] vocabularies for device description. The structure of the semantic device description is divided into four modules connected to the core ontology concepts:

- Device capabilities (hardware, software properties and state machines)
    - Semantic description of device capabilities (hardware platform model, software platform model, state machine)
- Device services

- presents the semantic description of device services on the abstract level, based on OWL-S specification, service model enables the interoperability between devices and services, employing the service capabilities and input/output parameters, devices in HYDRA are provided with semantic descriptions by combining the device ontology with the SAWSDL standard for annotating device WSDL files

- Device malfunctions
  - represents possible errors that may occur on devices, it is described by the error code and the human readable name information
- Device security properties

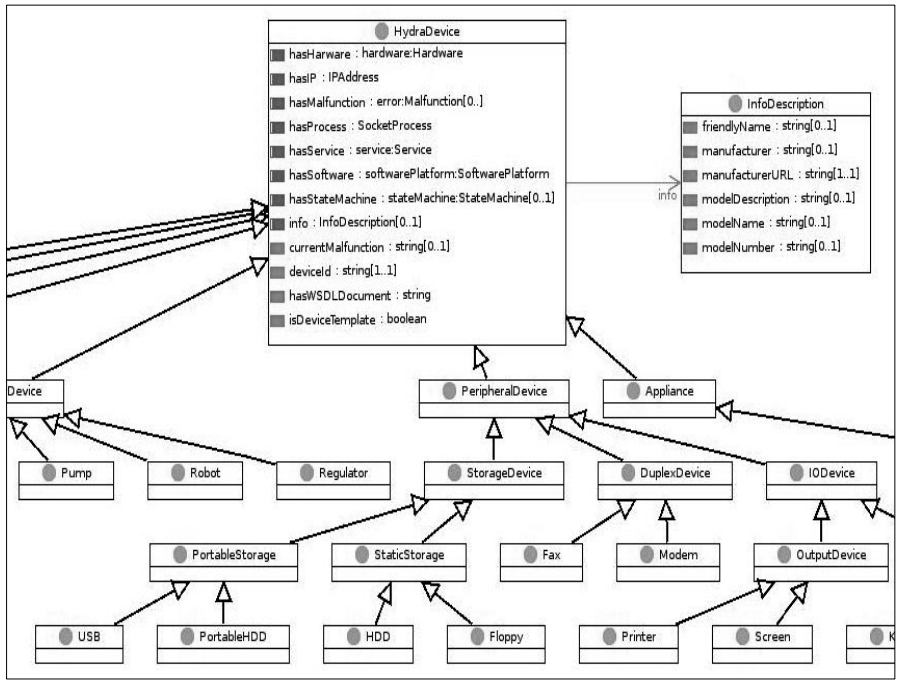


Fig. 1 Security properties attached to the devices

Semantic Security

Hydra should provide a vocabulary for protection goals and capabilities to cover existing devices and applications. It was assumed, that it has to describe a common model for semantic descriptions of different entities with different capabilities. Several ontologies used for security purposes were described and studied. NRL ontology [2] and MAMD (Multi Agent Multi Domains) security ontology [6] were

selected among various others, which in particular have met the needed requirements. According to Hydra needs and requirements, the NRL security ontology is the one, that fits mostly for Hydra and it was selected as a basis, the starting point to design the Hydra Security ontology.

The Hydra security ontology is a knowledge base representing the relation between different Protection goals and other security concepts, security mechanisms, algorithms and cryptographic primitives. Protection goals are high-level descriptions of security properties which have to be achieved by a system (for example confidentiality, non-repudiation, authentication, etc.). Security concepts then refer to commonly used methodologies at an abstract layer like various used algorithms, particular credentials, key store, etc. Security mechanisms describe directives how different security concepts are used in combination in order to achieve one or more protection goals. Those security mechanisms are usually based on a number of cryptographic primitives of different classes (encryption, hashing, signing, etc.).

With respect to the proposed Hydra Security Meta-model, the purpose of the proposed Protection Goals matches the *SecurityObjective* concept in the NRL Security Ontology. In general, *SecurityObjective* class enables to specify security objectives for the *SecurityConcept* class using the *supportsSecurityObjective* property. *SecurityObjective* also enables users to search for protocols, mechanisms, or policies based on the security objective (*Protection goal*) they require.

Usage of security ontology in Hydra assumes the connection between the device description and the security ontology. Each device has to be described also with security properties; therefore there is a need to link between the device ontology and security ontology. The main ontology class covering the most of basic security concepts, called *SecurityConcept* was linked with the main Device Ontology concept *HydraDevice* and the main service ontology concept *Service*. This simple interconnection of ontology concepts enable to easily add the any security properties to the both devices and services separately. For illustration of security properties representation, see Figure 2.

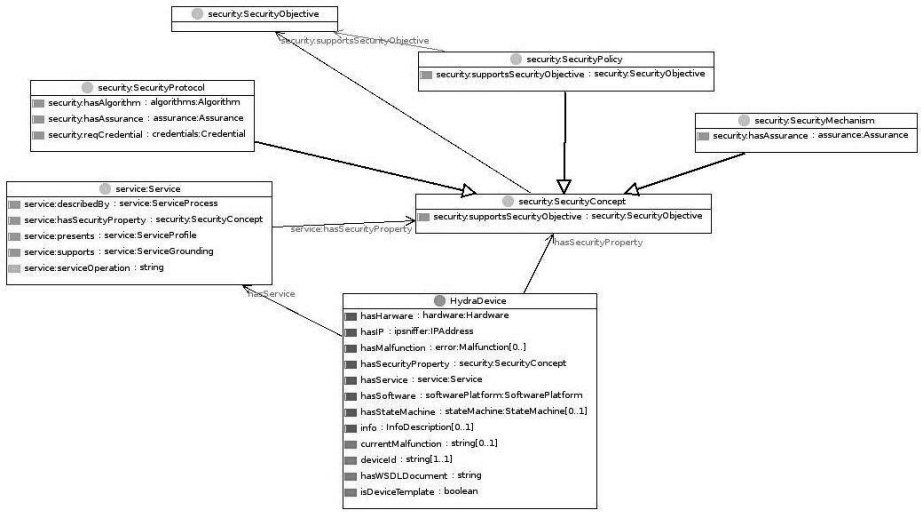


Fig. 2 Security properties attached to the devices

Conclusions

This paper presented the implementation of semantic technologies within the Hydra project. It briefly overviewed the basic concepts of semantic technologies usage within the middleware in both modes – design-time and run-time. Semantic devices and semantic description of such devices are then described; its extension with security related concepts is explained in more detail.

Acknowledgements

The work presented in the paper is supported by the EC within the FP6 IST-2005-034891 Project "HYDRA – Networked Embedded System Middleware for Heterogeneous Physical Devices in a Distributed Architecture"; by the Slovak Research and Development Agency under the contract No. APVV-0391-06 and by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/3135/06.

References

1. Kostelnik, P., Sarnovsky, M., Hreno, J., Rosengren, P., Ahlsen, M., Kool, P., Axling, M. : Semantic Devices for Ambient Environment Middleware, to be published in proceedings of Euro TrustAMI 2008, 15-18.9

2. Naval Research Lab. NRL security ontology http://chacs.nrl.navy.mil/projects/4SEA/ontology.html, 2007.2. Bruce, K.B., Cardelli, L., Pierce, B.C.: Comparing Object Encodings. In: Abadi, M., Ito, T. (eds.): Theoretical Aspects of Computer Software. Lecture Notes in Computer Science, Vol. 1281. Springer-Verlag, Berlin Heidelberg New York (1997) 415–438
3. HYDRA: Networked Embedded System middleware for Heterogeneous physical devices in a distributed architecture", Project Proposal, September 2005
4. FIPA Device Ontology Specification, Foundation for intelligent physical agents, 2002.
5. Amigo middleware core: Prototype implementation and documentation, deliverable 3.2. Technical report, Amigo Project, IST-2004-004182, 2006
6. FIPA Security Ontology Specification, available online: http://www.elec.qmul.ac.uk/staffinfo/stefan/fipa-security/

Authors Index

František Babič, Ján Paralič, Andreas Rauber
Editors

Proceedings

8th International Student Workshop WDA 2008

1st edition, 100 copies, Published by EQUILIBRIA, s.r.o.
for Centre for Information Technologies,
Faculty of Electrical Engineering and Informatics,
Technical University in Košice, Slovakia

Printed by EQUILIBRIA, s.r.o.

2008

ISBN 978-80-89284-21-4