Significance of Clustering and Classification Applications in Digital and Physical Libraries

Ioannis Triantafyllou[1,a)], Alexandros Koulouris[1,b)], Spiros Zervos[1,c)], Markos Dendrinos[1,d)], Daphne Kyriaki-Manessi[1,e)], Georgios Giannakopoulos[1,f)]

[1]*Technological Educational Institute of Athens*
Department of Library Science and Information Systems
Agiou Spyridonos Str., 12210, Aegaleo, Athens, Greece

[a)]triantafi@teiath.gr, [b)]akoul@teiath.gr, [c)]szervos@teiath.gr,
[d)]mdendr@teiath.gr, [e)]dkmanessi@teiath.gr, [f)]gian@teiath.gr

Abstract. Applications of clustering and classification techniques can be proved very significant in both digital and physical (paper-based) libraries. The most essential application, document classification and clustering, is crucial for the content that is produced and maintained in digital libraries, repositories, databases, social media, blogs etc., based on various tags and ontology elements, transcending the traditional library-oriented classification schemes. Other applications with very useful and beneficial role in the new digital library environment involve document routing, summarization and query expansion. Paper-based libraries can benefit as well since classification combined with advanced material characterization techniques such as FTIR (Fourier Transform InfraRed spectroscopy) can be vital for the study and prevention of material deterioration. An improved two-level self-organizing clustering architecture is proposed in order to enhance the discrimination capacity of the learning space, prior to classification, yielding promising results when applied to the above mentioned library tasks.

Keywords. Classification, Clustering, SOM, Digital Libraries, Summarization, Document Routing, Ontologies, Query Expansion, Paper-based Libraries, FTIR, Material Deterioration.

1. INTRODUCTION

Classification comprises the kernel mechanism of any document retrieval, routing or filtering model. That is, given a set of two or more classes, a classification system assigns each document to the appropriate class(es), as well as estimating a degree for each particular assignment.

Classification is also an essential procedure in physical (paper-based) libraries. Traditional classification systems, like Dewey Decimal Classification (DDC) are used in many physical libraries around the globe. In the digital environment, classification is changing. In repositories traditional classification systems are still in use. However, flexible keyword based classification systems like ontologies and tags are rapidly used by information professional and users. Self archiving has imported the use of tagging in repositories, in blogs, in social media, etc. The integrated approach of information is the future. Most interoperability issues concerning the information integration that is in different forms and media. Automated harvesting is applied in collective schemas like Europeana. In this sense, the applications that are proposed in this paper will be useful in document clustering and classification in harvesting procedures and will supplement other interoperability efforts in solving aggregation problems. Libraries and especially digital ones can benefit from this, because they will import and export content that will be automatically classified in subject categories, in a batch and automated way during the harvesting.

2. CLASSIFICATION/CLUSTERING SYSTEM

A number of text classification approaches have been presented and extensively tested in the past. The most common ones are based on linear text classification models, k-nearest-neighbor algorithms, bayesian independence classifiers, neural networks and support vector machines. Extensive evaluation of these techniques can be found in [17][35]. However, all methods presented so far are applied directly to the training space without any attempt to examine and possibly improve its discrimination capacity. We argue that by clustering training sequences we can significantly improve discrimination capacity, thus improving overall performance. Self-Organizing Maps (SOMs) have been proposed as the main clustering model. A project that aims at constructing methods for exploring full-text document collections [27][32][33] sprung from the suggestion of using SOMs as a preprocessing stage for encoding documents. These maps are used to automatically cluster documents according to the features that they contain. Documents are organized, during a preprocessing stage, based on a map, in such a way that similar documents are projected into nearby locations. This type of ordering facilitates the extraction of clusters by using intuitive neighborhood relations.

Even though the proposed classification system was originally designed to handle document collections it can easily applied to other type of data collections, i.e. FTIR scanning measurements from books in order to identify the stages of paper deterioration [3.2].

2.1. System Architecture

The overall architecture of the proposed text classifier is divided into two levels (Fig.1). At the first level a SOM has been trained aiming at the organization of the entire document collection into smaller clusters. Clustering is based on the classes of the documents rather than the documents themselves.

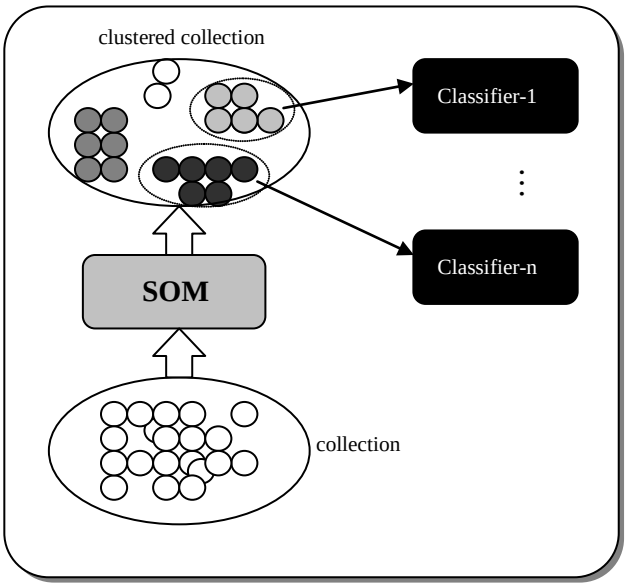

FIGURE 1. Overall architecture of the proposed classifier

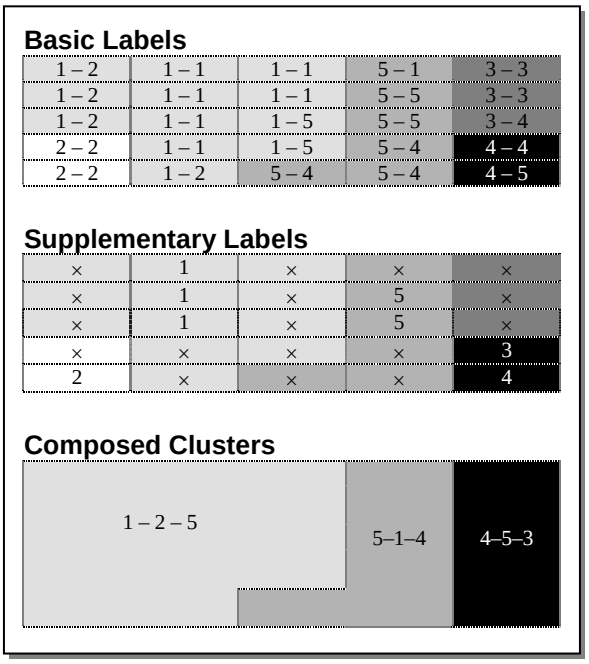

FIGURE 2. A clustering example

In this respect, clusters contain classes with related class-labels. Each new document is first filtered by the trained SOM and assigned to a specific cluster of classes. Each cluster is associated with a respective traditional classifier trained over this particular cluster. Since each traditional classifier deals with only a part of the training space, more specifically the number of classes belonging to the respective cluster, discrimination capacity is improved. As a traditional classifier we can use any of the well established classification models: Simple Linear, kNN, Bayesian, SVM, etc.

2.2. SOM (Self-Organizing Map)

The SOM is a method for producing ordered low-dimensional (usually two-dimensional) representations of an input data space [31]. Typically such input data is complex and high-dimensional with data elements being related to each other in a nonlinear fashion.

Two main properties of the SOM, indicating that it would constitute a very efficient clustering model, are [25]:

1. Approximation of the Input Space. The produced map represented by the set of synaptic weight vectors provides a good approximation to the input space.
2. Topological Ordering. The produced map computed by the SOM algorithm is topologically ordered in the sense that the spatial location of a neuron in the lattice corresponds to a particular domain or cluster of input patterns.

2.2.1. Training Phase

The algorithm responsible for the formation of the SOM starts by randomly initializing the synaptic weights in the network. Three essential processes run iteratively until the map converges:

1. Competition. For each input pattern the neurons in the network compute their respective values of a discriminant function (inner product) and the neuron with the largest value is declared winner.
2. Cooperation. The winning neuron determines the spatial location of a topological neighborhood of excited neurons.
3. Synaptic Adaptation. The last mechanism enables the excited neurons to increase their individual values in relation to the input pattern through suitable adjustments.

Let n denote the dimension of the input space, $\underline{x}(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T$ denote the input vector selected at time t, m denote the number of total neurons in the produced map and $\underline{w}_j(t) = [w_{j1}(t), w_{j2}(t), \ldots, w_{jn}(t)]^T$ denote the weights for node j at time t. The winning node s is selected so that,

$$\|\underline{x}(t)-\underline{w}_s(t)\| = \min_{j=1,2,\ldots,m} \|\underline{x}(t)-\underline{w}_j(t)\|$$

After finding the winning node, adaptation of the weights of the nodes within a defined neighborhood is performed as shown,

$$w_{ji}(t+1) = w_{ji}(t) + a(t)\,[x_i(t) - w_{ji}(t)],\ i=1,2,\ldots,n$$

where a(t) is a gain term. The a(t) is selected to guarantee that the SOM converges by introducing two control mechanisms. The first one is that a(t) decreases in time and converges to 0 allowing the SOM to converge. The second one is that, a(t) implicitly defines the neighborhood area of the winning node. A gain term including both of the previous mechanisms [25], is adopted :

$$a(t) = A1 \times \exp(-t / A2) \times \exp(-\|r_j-r_s\|^2 / 2\sigma^2(t)),$$

where $\sigma(t) = A3 \times \exp(-t / A4)$, r_j and r_s define the position of excited neuron j and the position of winning neuron s respectively, and finally $A1,A2,A3,A4 \in \Re$.

2.2.2. Producing Clusters of Classes

After constructing the map, nodes are labeled and then clusters are composed based on these labels (Fig.2). The labeling process, called "simulated electrode penetration mapping" [25][39], consists in finding for each neuron the most related input vector (the input vector for which the neuron produced the best response) and assigning the respective label to this neuron. In order to further assist the clustering process we assign two labels to each neuron, the labels of the two most related input vectors, which we call basic labels. In addition, supplementary labels are assigned to the neurons as follows: for each input vector, the neuron with the strongest response is assigned to the respective input label. Some neurons are left without supplementary labels. The clustering process consists of three subsequent phases. At the first phase, areas are spotted based on the first basic label. Area continuity is ensured by the cooperative behavior of neighboring neurons. Then, each area is enriched with all second basic and supplementary labels. Finally, any adjacent areas one being a subset of the other are merged, iteratively, until no other merging is possible. Overlapping between final clusters is allowed, meaning that certain labels can appear in more than one cluster.

2.3. Traditional Classifiers

2.3.1. Linear Classifier

A linear text classifier represents a class c as a weight vector

$$\underline{w}(c)=[w_{f1}(c),w_{f2}(c),\ldots, w_{fn}(c)]^T,$$

where $f_{j,\ j=1,2,\ldots,n}$ is the set of all features and n is the size of this set (the dimension of the input space). Decision is made through evaluating the score of a document d with every class c, by computing the dot product:

$$F_c(d) = \underline{s}(d) \bullet \underline{w}(c),$$

where $\underline{s}(d) = [s_{f1}(d), s_{f2}(d),\ldots, s_{fn}(d)]^T$ is the strength of each feature in document d.

Multiplicative weight-updating algorithms such as Winnow [40] have been studied extensively in the theoretical learning literature. They perform exceptionally well in domains with very high dimensionality and particularly in the presence of irrelevant attributes, noise [17]. During the training phase of multiplicative weight-updating algorithms, adaptation of weight vector is mistake-driven. If the algorithm predicts 0 (no assignment in this class) and the correct value is 1, then we have a positive example and the weight vector is promoted: multiplied by a factor a, where a>1. If the algorithm predicts 1 (assignment in this class) and the correct value is 0, then we have a negative example and the weight vector is demoted: multiplied by a factor b, where 0<b<1. The adopted linear classifier is the one referred to as Balanced Winnow. In this case the algorithm keeps two weight vectors $\underline{w}^+$, $\underline{w}^-$. The overall weight vector is then computed as the difference between these two vectors: $\underline{w}=\underline{w}^+-\underline{w}^-$. During training, a positive example alerts promotion to the positive weight vector and demotion to the negative weight vector, while a negative example alerts demotion to the positive weight vector and promotion to the negative weight vector.

Multiclass cases are handed either by the one-versus-all solution or by one-versus-one solution. In the first approach the classifier with the highest output assigns the class. In the second approach, classification is done by a max-wins voting strategy, in which every classifier assigns the instance to one of the two classes for every possible combinations.

2.3.2. kNN Classifier

The most commonly used learning technique is the k-nearest neighbor (kNN). Let $\underline{x}$ denote the input vector, n denote the dimension of the input space and $\underline{x}_1,\underline{x}_2,\ldots,\underline{x}_k$ denote its k-nearest neighbors. Nearest neighbors are defined in terms of the standard Euclidean distance,

$$d(\underline{x},\underline{x}_i) = \sqrt{\sum_{d=1,2,\ldots,n}(x^d - x_i^d)^2}$$

If y is the function that maps each known vector to a specific class, then $\hat{y}$ is the estimate function of y. One obvious refinement to the kNN algorithm is the distance-weighted kNN algorithm,

$$\hat{y}(\underline{x}) = \underset{c\in C}{\operatorname{argmax}} \sum_{i=1}^{k} w_i\, \delta(c,y(\underline{x}_i))$$

where $\delta(a,b)=1$ if a=b, otherwise $\delta(a,b)=0$, $w_i = 1/d(\underline{x},\underline{x}_i)^2$, and C is the set of all classes.

The adopted text classification model was based on the distance-weighted kNN algorithm. One practical issue in applying kNN algorithms is that the distance between instances is calculated based on all their attributes. As less attributes are relevant to the classification process, the more misleading assignments of class labels occur. This difficulty, which arises when many irrelevant attributes are present, is sometimes referred to as the curse of dimensionality [45]. One way to overcome this problem is rejecting as many irrelevant attributes as possible. This is achieved by clustering similar instances.

2.3.3. Bayesian Networks

Bayesian networks (BNs), belong to the family of probabilistic graphical models. These graphical structures are used to represent knowledge about an uncertain domain. In particular, each node in the graph represents a feature/class, while the edges between the nodes represent probabilistic dependencies among the corresponding features/class. These conditional dependencies in the graph are often estimated by using known statistical and computational methods. Naive Bayesian networks (NB) are very simple BNs which are composed of directed

acyclic graphs with only one parent (representing the unobserved node-class) and several children (corresponding to observed nodes-features) with a strong assumption of independence among child nodes in the context of their parent.

Given a class variable C, the feature variables $F_1,\ldots,F_n$ and using the Naïve Bayes's induction we can estimate

$$P(C \mid F_1,...,F_n) = \frac{P(C)P(F_1,...,F_n \mid C)}{P(F_1,...,F_n)} \propto \frac{P(C)\prod_{i=1}^{n} P(F_i \mid C)}{P(F_1,...,F_n)}$$

and the NB classifier is defined as follows

$$Classify(f_1,...,f_n) = \underset{\forall c \in SetOfClasses}{ArgMAX}\ P(C=c)\prod_{i=1}^{n} P(F_i = fi \mid C=c)$$

A more efficient alternative to NB is the Averaged N-Dependence Estimators (A1DE-A2DE). It was developed to address the attribute-independence problem of the popular NB classifier and achieves highly accurate classification by averaging over all of a small space of alternative NB-like models that have weaker (and hence less detrimental) independence assumptions than NB. The resulting algorithm is computationally efficient while delivering highly accurate classification on many learning tasks [61].

2.3.4. SVM (Support Vector Machine)

The aim of a linear SVM classifier is to find the maximum-margin hyperplane, the one with the largest separation, between the two classes (Positive/Negative). If such a hyperplane exists and the training data are linearly separable then a pair (**w**,b) exists such that

$$w \bullet x - b \geq 1\ ,\quad \forall x \in P \quad \text{and} \quad w \bullet x - b \leq -1\ ,\ \forall x \in N$$

and the classifier assigns based on the sign of the dot product between the weight vector w (hyperplane) and the new point(case) excluding the area of the margin threshold b: $classify_{w,b}(x)=sign(w \bullet x-b)$.

It is easy to show that the optimization problem that we have to solve is the quadratic programming problem

$$\arg\min_{(\mathbf{w},b)} \frac{1}{2}\|\mathbf{w}\|^2 .$$

Most real-world problems involve non-separable data for which no hyperplane exists that successfully separates the positive from negative instances in the training set. [16] suggested a modified maximum margin idea in order to handle data contains misclassified instances introducing a soft margin allowing some misclassifications during the training phase. This is achieved by adopting the non-negative slack variables ξ_i , which measure the degree of misclassification of the data x_i . The new soft margin constraint along with the objective to minimize ||w|| can be solved using Lagrange multipliers concluding to the following optimization problem

$$\arg\min_{\mathbf{w},\xi,b}\ \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i[y_i(\mathbf{w}\cdot\mathbf{x_i} - b) - 1 + \xi_i] - \sum_{i=1}^{n}\beta_i\xi_i \right\}$$

with $a_i, b_i \geq 0$.

For a nonlinear classification problem [10] suggested the kernel trick originally proposed by [1]: mapping the data to some other, possibly infinite dimensional, Hilbert space H as Φ: $R^d \rightarrow H$. If there were a kernel function K such that, $K(x_i,x_j)=\Phi(x_i) \bullet \Phi(x_j)$ we would only need to use K in the training algorithm, and would never need to explicitly determine Φ. Thus, kernels are a special class of function that allow inner products to be calculated directly in feature space, without performing the mapping described above [53]. Some most well known nonlinear kernels are:

Polynominal => $K(x_i,x_j) = (x_i \bullet x_j+1)^d$

Gaussian => $K(x_i,x_j) = \exp(-\gamma \|x_i-x_j\|^2)$, where usually $\gamma=1/2\sigma^2$

Tangent => $K(x_i,x_j) = \tanh(k\, x_i \bullet x_j-\delta)^d$, where $k,\delta >0$

SVM's are capable to deal with classification problems where the number of features is large with respect to the number of training samples. The data points that lie on the margins are known as support vector points and the solution is represented as a combination of only these points and the other data points are ignored. Therefore, the model complexity of an SVM is unaffected by the number of features encountered in the training data. Multiclass cases are arranged by using the approaches already described in 2.3.1.

2.4. Feature Selection

Special consideration has been given to the selection of the feature space, a crucial aspect in the performance of any text classification model as well as of the results of the SOM training phase. Any n-word in the training corpus constitutes a candidate feature. Functional words are excluded based on stop-lists. Feature selection is further supported by a word conflation tool. The tool aims at capturing morphological variations of words located in the document collection, through a process called "conflation" [22].

2.4.1. Information Gain (IG) - Average Mutual Information (AMI)

Feature selection consists in reducing the vocabulary size of the training corpus by selecting n-words with the highest mutual information over the set of values of the class variable. Information Gain (or Average Mutual Information) is the difference between the entropy of the class variable, H(C), and the entropy of the class variable conditioned on the presence or absence of the feature (word or phrase) variable, H(C|F). This method has been adopted by several researchers [62][28][44].

Let C denote the random variable over all classes and F the random variable over the presence or absence of feature f in a document, the Information Gain is given by the following formula:

$$\mathbf{IG(C|F)} = \mathbf{H(C) - H(C|F)} = -\sum_{c\in C} P(c)\log(P(c)) + \sum_{f\in\{0,1\}} P(f)\sum_{c\in C} P(c\mid f)\log(P(c\mid f)) =$$

$$= \sum_{c\in C}\sum_{f\in\{0,1\}} P(c,f)\log\left(\frac{P(c,f)}{P(c)P(f)}\right)$$

where, P(c) is the number of documents with class label c divided by the total number of documents, P(f) is the number of documents containing (or not) the feature f divided by the total number of documents and P(c,f) is the number of documents with class label c that also contain (or not) feature f, divided by the total number of documents.

2.4.2. TF.IDF

Feature selection is also performed based on the TF.IDF measure [38][28]. In our case we have classes rather than documents, so TF is the frequency of feature "i" within a class c, and IDF is the logarithm of N/N_f, where N is the total number of classes and N_f is the number of classes containing the feature "i".

$$tf_i * idf_i = TF_i \cdot \log\frac{N}{N_i}$$

The selected features are the most dominant ones in each class. Furthermore, the number of selected features for each class is related to the frequency of the class.

2.4.3. Proposed Selection

By evaluating the previously described methods we have concluded that the IG measure outperforms the TF.IDF measure. Furthermore, the final selection of dominant features per class, was based on a combination of the previous measures, namely, the product of IG and TF.IDF. The combined measure improved the results of the SOM training phase. Once again, as in the case of the TF.IDF measure, the selected features are the most dominant ones in each class and the number of selected features per class is also related to the frequency of the class.

2.5. System Evaluation

To provide an objective basis of comparison, we used the document collection of the Reuters newswire of 1987, properly identified as Reuters-22173 [4], but hereafter referred to as Reuters. We chose to experiment with 3965 documents as training cases and 1056 documents as test cases. There were 57 class labels (topics) of interest, occurring more than twice in the training data. Stories with no class labels were left out of the process.

The first level consisted in training the self-organizing map. The number of neurons (N_n) we used was related to the average size of the classes contained in the training set, $N_n \simeq N_c \sqrt{D / N_C}$, where N_c is the total number of classes and D is the total number of documents in the training set. Based on our training set we estimated a value of $N_n = 486 \simeq 475.399$ [59].

Feature selection was based on the combined IG × TF.IDF measure [59]. The dimension of the feature space is an important parameter of the training phase of the SOM. Previous tests led to a 420-dimensional feature space [59].

Performance results [59] were enriched with new traditional classifiers (bayes, svm) showing improvement (Table 1) of the classification accuracy when the suggested clustering of the training space prior to classification was applied.

TABLE 1. Efficiency of the Evaluated Classifiers

Method \ Classifier	Linear	kNN	NaiveBayes	A1DE	SVM
Traditional Classification	84%	84%	78%	87%	90%
SOM Clustering prior to Classification	88% (+4%)	90% (+6%)	86% (+8%)	91% (+4%)	93% (+3%)

3. LIBRARY APPLICATIONS

Applications of clustering and classification techniques can be proved very significant in both digital and physical (paper-based) libraries. The most essential application, document classification and clustering, is crucial for the content that is produced and maintained in digital libraries, repositories, databases, social media, blogs etc., based on various tags and ontology elements, transcending the traditional library-oriented classification schemes. Other applications with very useful and beneficial role in the new digital library environment involve document routing, summarization and query expansion. Paper-based libraries can benefit as well since classification combined with advanced material characterization techniques such as FTIR (Fourier Transform InfraRed spectroscopy) can be vital for the study and prevention of material deterioration.

3.1. Digital Library Applications

3.1.1. Document Classification/Clustering

Classification is an essential library oriented procedure in physical libraries. After cataloguing, classification directs the users to discover the information they need according to the subject. Subject analysis of information and classification are two parallel procedures. The subject analysis defines the subject of a document and the classification defines the category. Especially in paper-based libraries, most of the classification schemes are the arithmetic approach of the subject, for example the Dewey Decimal Classification (DDC). DDC number is the subject category of the document.

But things are changing rapidly with the use of technology in libraries. The classification applies in digital libraries, in repositories, in blogs, wikis, Facebook, Twitter, etc. The traditional classification schemes are still in use in digital libraries, but they are supplemented of the forms of classification and clustering. For example, ontologies and tags are new forms of classification that are based in keywords and are very flexible for users and information professionals. WEB 2.0 has imported new rules in information classification and clustering. For example, in blogs clustering is based on tags. The same procedure is followed in YouTube and in other social media. Users tagging

and annotation, is an added value tool that can be used for information discovery and classification in Facebook and other social media.

Document classification and clustering is very important in digital libraries and repositories. Repositories are not isolated. Repositories are exchange information, either in national or in international level. Synergies like Europeana, the European Digital Library, Archive and Museum [34] uses metadata harvesting in an automated way. Tools and application that are proposed in this paper can be fully implemented in such information collective schemas. The concept of the document clustering and classification is the use of subjects, keywords and tags that are predefined and can be extended from any user. For example, if the National Archive of PhD Theses [48] in Greece that is maintained by the National Documentation Centre (EKT), imports theses from the Greek Universities the use of document classification application, will in an automated way import the theses and the appropriate collection according to the subject, by using an automated batch process. This will solve many interoperability problems that are emerge in the automated harvesting [24].

3.1.2. Document Routing

Electronic document routing is the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant, human or machine to another for action, according to a set of procedural rules. In digital libraries can be applied in new material automatic or semi-automatic (with human approval) classification and in retaining updated information through users notifications on material that suites its needs or favorites (Fig.3). In the first case it performs a typical classification process as described in 3.1.1 for the new material arriving at the DL. It also can be applied in filtering new material harvesting from other web sources. Combined with document summarization [3.1.3] it can be operated as an automatic RSSfeed-like tool for the registered users of the DL. Content of new material can be scanned continuously searching for new matches of the user needs, then transmits the content summary of these matches by way of feeding the information to subscribers.

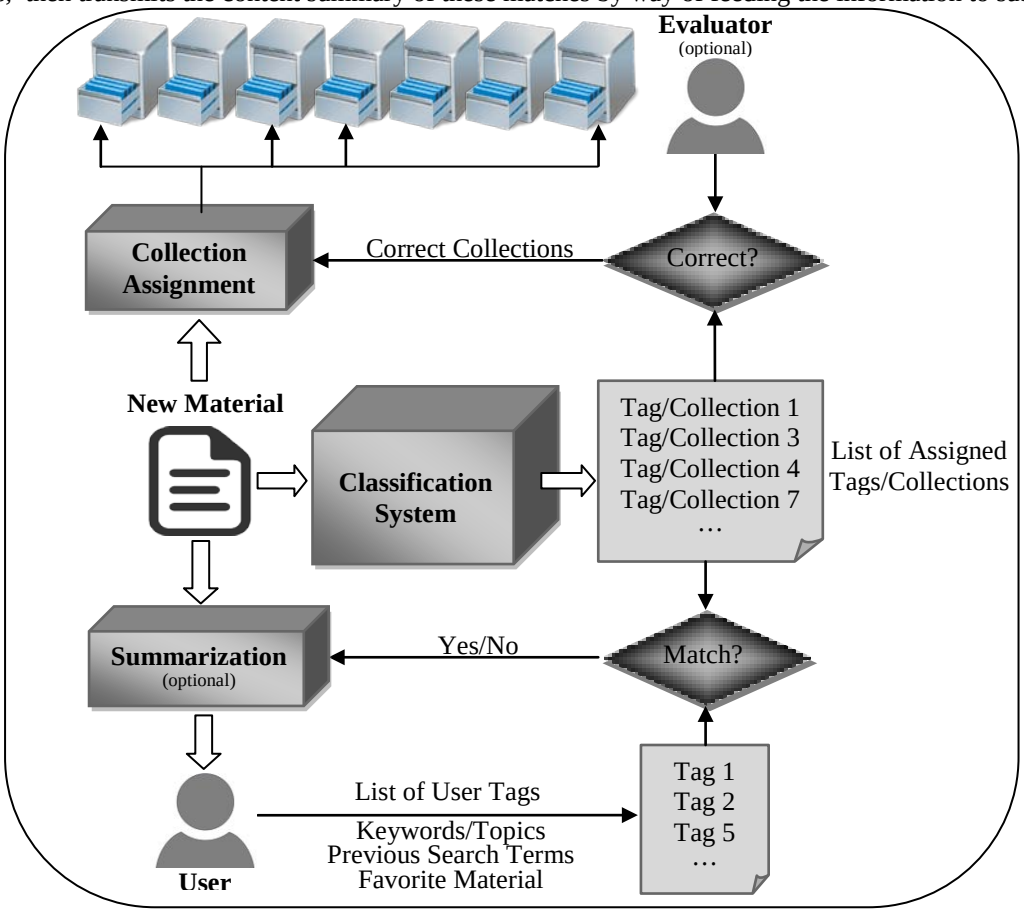

FIGURE 3. Document Routing in a DL Environment

3.1.3. Document Summarization

In an increasingly information-laden world where unstructured text data are the predominant type of data stored online, systems that can filter and condense data so that only relevant information reaches the decision maker, have become the focus of considerable interest and investment. A summary can be defined as a synopsis of the content of a document by distilling the most important information for a particular user and task. Early systems were characterized by a shallow approach such as exploiting term frequency [42] or cue and location features [20]. Corpus-based systems follow up classical approaches. They combine the calculation of corpus statistics in a learning framework. [36] has developed a Bayesian classifier, [47] combines individual features with the Dempster-Shafer rule while [3] combines features by the Bayesian rule.

Earlier summarization systems [18] aim to extract the most important sentences of the text by a set of features incorporate shallow linguistic processing for term extraction and statistical filtering through a general corpus. A reductive transformation of the source text to summary by sentence selection rather than a full understanding of the text by parsing to logical form or the exploration of its discourse structure, is an adequate framework to apply and evaluate summarization systems. We focus on environments, as DL, where indicative, information retrieval oriented summaries are useful, bearing in mind that without an intermediate source processing and possibly a full text interpretation, part of the important content might be missed.

The final extraction of the selected sentences in any summarization systems is supported by a machine learning classification system dividing the sentences in 2 major categories, positive and negative ones, assign them a confidence score. Most relevant (high-scored) sentences are then selected to appear in the summary.

3.1.4. Query Expansion

Classification benefits can also contribute to relevance feedback and query expansion techniques. One way to improve effectiveness is to better represent the information need by adding useful terms to the query. Proper weighting of terms allows expansion of the query by adding new terms for which we have evidence of usefulness. The problem that adds to the complexity is to distinguish between good terms and poor terms: good terms tend to co-occur non-randomly within the relevant documents (as opposed to the rest of the collection) while poor terms tend to co-occur randomly. Which are the relevant documents? If the feedback is specified by the searcher then we have a confident evidence of relevance. Otherwise, the idea of treating the top resulted documents as being relevant, had been examined in the past [11]. A SOM-based improved architecture for query reconstruction using statistical evidence of the underlying document collection has been introduced yielding enhanced query results [60].

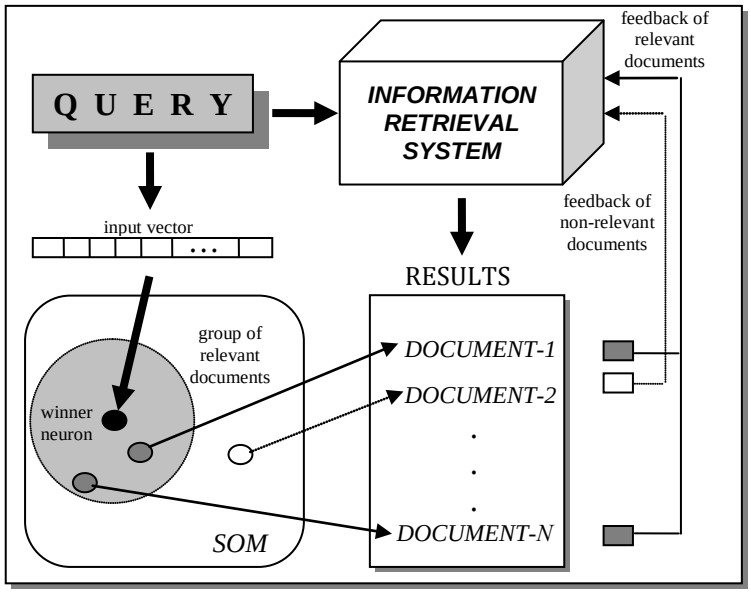

FIGURE 4. Proposed architecture for reconstruction of the query using a self organizing map. Grey nodes/boxes indicate relevant documents while white ones indicate irrelevant. Decision is based on a KNN area around the winner neuron.

Other techniques are aiming at constructing user query profiles [41]. One may use domain-specific ontology, such as thesauri or concept hierarchies, to sharpen a user query, or use an interactive dialog interface program that pops up a dialog screen for users to enter meanings or synonyms whenever the user poses a query on the fly.

An ontology scheme in general might also improve the results of a full text search or a search in a database environment. The ontology consists of a hierarchy of classes and subclasses, as well as a set of freely defined relations between classes [37]. Let consider as a case study an educational ontology with the following classes: institute, department, professor (defined in all its ranks), library, book, copy, student, as well as all the connecting relations. A search in a document database through a certain keyword professor will retrieve not only the documents with this word, but also all other documents with terms hierarchically related with professor, such as lecturer (subclass of professor). Another retrieval extension occurs based on the relations defined between the given keyword class and other terms (classes), comprising also the documents containing the term book, due to the relations 'professor borrowed copy' and 'copy of a book'. Similarly, searching documents with the keyword institute will result to an extended retrieval, consisting not only the documents with the term institute, but also documents with the term library, due to the relation 'library installed_in institute'.

The ontology is usually stored in an rdf/owl file, through rdfs codes for labeling either the hierarchical position of each class or the defined properties (relations) between classes. Given the search keyword, it is initially sought within the ontology file. If it is located there, next it is examined whether there are other classes connected to the initial one though the owl label owl:subClassOf or owl:objectProperty within the description segment of the keyword. In this case, the starting search keyword is extended to a set of terms, thus increasing the recall of the retrieval.

3.2. Paper-based Library Applications

Traditional information substrates such as paper, kept in libraries, archives and museums are very complex materials. Paper consists mainly of cellulose, but depending on the source of the raw materials used for its production and the papermaking process may also contain lignin, hemicelluloses, fillers, sizing materials, colorants, metal ions etc [51]. To make matters worse, the deterioration process alters the initial chemical composition of most of paper components by introducing new functional groups to cellulose, lignin and hemicelluloses and by producing new chemical species [65].

The characterization of paper and the estimation of its condition is a very important aspect of preservation in libraries, archives and museums, because it forms the basis for preservation-related decision making. It is traditionally accomplished by various tests which are tedious, time consuming and very often destructive [57].

FTIR spectroscopy is a micro-destructive technique for material characterization which has been often used for paper characterization. An FTIR spectrum is essentially a feature-rich plot of the absorption intensity of infra-red light absorbed by the material against the frequency (wavenumber) of the light. IR radiation is absorbed by matter, and the frequencies of the absorption spectrum of a single chemical compound can serve as the compound fingerprint [49]. FTIR spectroscopy combined with an ATR (Attenuated Total Reflection) detector offers a non-destructive and reagent-free method allowing for the fast identification of a compound.

FTIR spectroscopy applications for paper characterization include:

- The study of paper ageing [7][54][30][2][43][50][56][12].
- The study of the mechanism of foxing, that is, the production of rust-coloured stains on paper [15], of the effects of pollution on paper [29] and of paper photoxidatuion [63].
- The identification of the low molecular products of paper ageing [19].
- The detection of functional groups produced by oxidation or existing in paper as part of its constituents (carbonyls, carboxyls, amines, conjugations, etc) [26][14][2].
- The evaluation of paper conservation treatments [13][46][66][64][58][6][52].
- The detection of lignin [55], gelatine [8][9][15][46] and various paper additives [23].

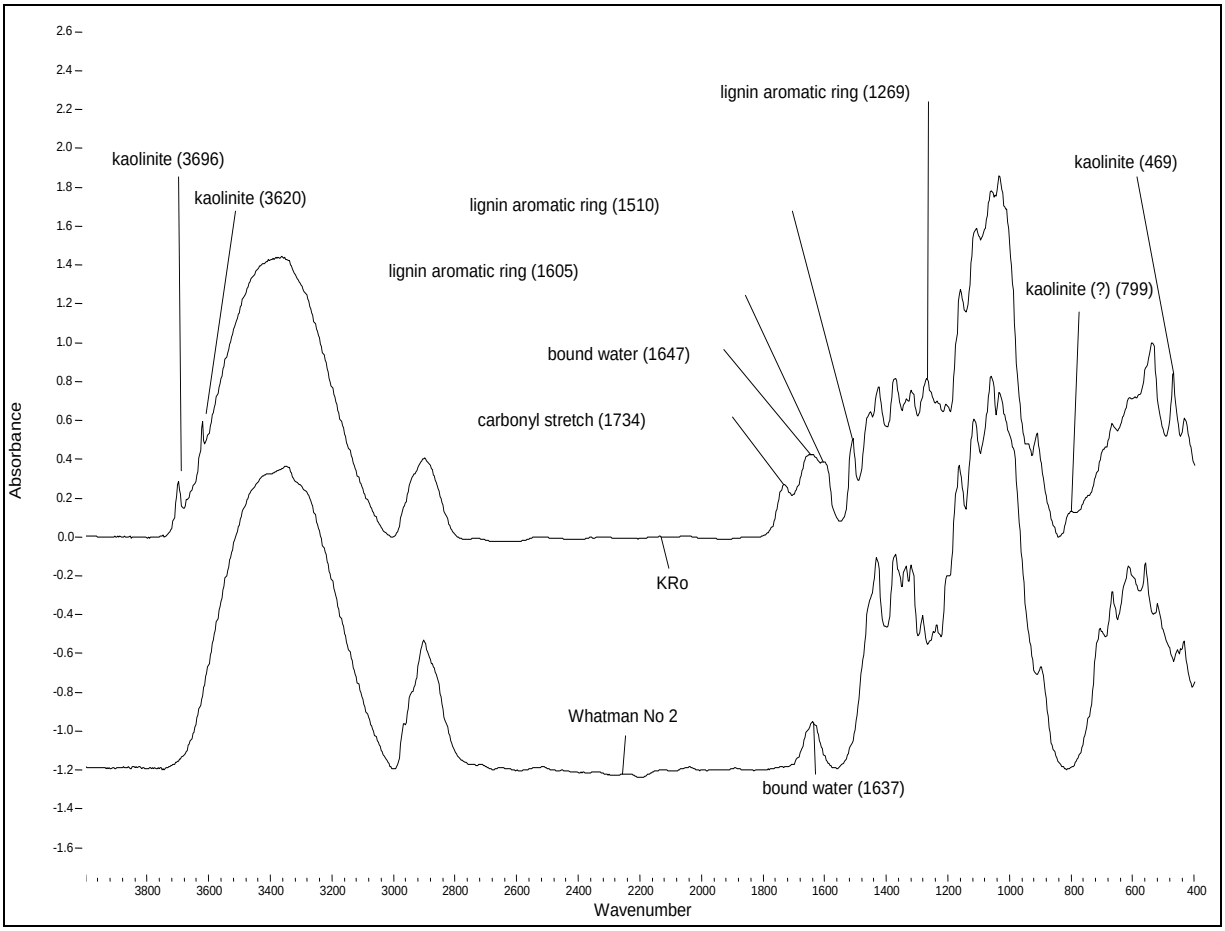

FIGURE 5. FTIR spectra of a pure cellulose paper (Whatman no 2) and a historic paper made from mechanical pulp (containing lignin) and with kaolinite filler (KRo)

The combination of FTIR spectroscopy equipped with ATR detector with classification methods has been used in various disciplines including food quality control with great success in predicting food microbial spoilage [21][5], by classifying it in two classes, that is, spoiled and unspoiled. In a similar manner, it can significantly automate paper characterization and provide information about its condition, its composition (presence of lignin), its additives and fillers (calcium carbonate, rosin, alum, gelatine) etc. For example, it can classify paper in three or more classes and automatically assist selection for conservation, especially selection for mass conservation treatments.

The three classes can be:

- Paper in good condition, strong and neutral or alkaline, not needing treatment
- Acidic paper having some strength left, needing only deacidification
- Acidic weak paper, needing deacidification and strengthening

Or in a different context, it can be used to discriminate between papers containing lignin, or having acidic pH, or having calcium carbonate filler, or gelatin sizing. These parameters can be combined in any desired manner and produce new classes. The combination of FTIR spectroscopy with classification methods can automatically predict if a certain paper falls within any of those classes.

4. CONCLUSION

This paper focuses on applications of clustering and classification techniques that can be proved very significant in both digital and physical (paper-based) libraries. The conceptual model and the architecture for these applications has been already developed. The most essential application that is proposed is the document classification and clustering, which is crucial for the content that is produced and maintained in digital libraries, repositories, databases, social media, blogs etc.. This application proposes classification and clustering techniques that are based on various tags and ontology elements, which transcending the traditional library-oriented classification schemes. Other applications with very useful and beneficial role in the new integrated information environment involve document routing, summarization and query expansion. These applications will supplement other solutions that has been proposed and implemented in distributed repository aggregations, for example in Europeana. Finally, paper-based (physical) libraries can benefit as well since classification combined with advanced material characterization techniques such as FTIR (Fourier Transform InfraRed spectroscopy) can be vital for the study and prevention of material deterioration. The improved two-level self-organizing clustering architecture that is proposed in order to enhance the discrimination capacity of the learning space, prior to classification, yielding promising results when applied to the above mentioned library tasks. Future tasks will include the implementation and the evaluation of the proposed architecture and tools.

5. REFERENCES

1. Aizerman Mark A., Braverman Emmanuel M., and Rozonoer Lev I. (1964). "Theoretical foundations of the potential function method in pattern recognition learning". Automation and Remote Control 25: 821–837.
2. Ali M., Emsley A.M., Herman H., & Heywood R. J. (2001). Spectroscopic studies of the ageing of cellulose paper. Polymer, 42, 2893-2900.
3. Aone C., Okurowski ME, Gorlinsky J. (1998). Trainable, Scalable Summarization Using Robust NLP and Machine Learning. In Proceedings of COLING-ACL 1998: 62-66.
4. Apté C., Damerau F., and Weiss S.M. (1994). Automated learning of decision rules for text categorization. ACM Transactions on Information Systems, 12(3):233-251, July 1994.
5. Argyri A.A., Panagou E.Z., Tarantilis P.A., Polysiou M.,& Nychas G.J.(2010). Rapid qualitative and quantitative detection of beef fillets spoilage based on Fourier transform infrared spectroscopy data and artificial neural networks. Sensors and Actuators B: Chemical, 145(1), 146-154.
6. Balakhnina IA, Brandt NN, Chikishev AY, & Rebrikova NL (2013). Effect of laser radiation on 19th century paper. Restaurator, 34(1), 30-44.
7. Banik G., & Ponahlo J. (1982/83). Some aspects of degradation phenomena caused by green copper-containing pigments. The Paper Conservator, 7, 3-7.
8. Barrett T. D. (1989). Early European papers/contemporary conservation papers - A report on research undertaken from fall 1984 trough fall 1987. The Paper Conservator, 13, 1-108.
9. Barrett T., Lang P., Waterhouse J., Cook J., Cullison S., Fuller B., . . . Pullman J. (1996). Non-destructive measurement of gelatin and calcium content of European papers: 1400 – 1800. Paper presented at the International Conference on Conservation and Restoration of Archive and Library Materials, Pre-prints, Erice.
10. Boser B.E., Guyon I.M., Vapnik V. N. (1992). "A training algorithm for optimal margin classifiers". Proceedings of the fifth annual workshop on Computational learning theory - COLT '92. p. 144.
11. Buckley C., Salton G., Allan J., Singhal A. (1994). "Automatic query expansion using SMART: TREC 3". 3rd Text REtrieval Conference (TREC3), Nov. 1994.
12. Calvini, P., & Silveira, M. (2008). FTIR analysis of naturaly aged FeCl3 and CuCl2-doped cellulose papers. e-Preservation Science, 5, 1-8.
13. Calvini, P., Grosso, V., Hey, M., Rossi, L., & Santucci, L. (1988). Deacidification of paper – a more fundamental approach. The Paper Conservator, 12, 35-39.
14. Caverhill, J., Stanley, J., Singer, B., & Latimer, I. (1999). The effect of aging on paper irradiated by laser as a conservation technique. Restaurator, 20(2), 57-76.
15. Choisy, P., De La Chapelle, A., Thomas, D., & Legoy, M. D. (1997). Non invasive techniques for the investigation of foxing stains on graphic art material. Restaurator, 18(3), 131-152.
16. Cortes C.; Vapnik V. (1995). "Support-vector networks". Machine Learning 20 (3): 273.

17. Dagan I., Karov Y., and Roth D. (1997). Mistake-driven learning in text categorization. In Second Conference on Empirical Methods in Natural Language Processing, 1997.
18. Demiros I., Antonopoulos V., Georgantopoulos B., Triantafyllou I., Piperidis S. (2001). "Connectionist models for sentence-based text extracts". IEEE-2001 (SMC), Volume 4, pages 2648-2653.
19. Dupont, A.-L. (1996). Degradation of cellulose at the wet/dry interface, II. An approach to the identification of the oxidation compounds. Restaurator, 17(3), 145-164.
20. Edmundson H.P. (1969). New methods in Automatic Extracting. In Journal of the Association for Computing Machinery 16(2):264-285.
21. Ellis, D. I., Broadhurst, D., Kell, D. B., Rowland, J. J., & Goodacre, R. (2002). Rapid and quantitative detection of the microbial spoilage of meat by Fourier transform infrared spectroscopy and machine learning. Applied and Environmental Microbiology, 68(6), 2822-2828.
22. Frakes W. B. (1984). Term conflation for information retrieval. Research and Development in Information Retrieval, New York: Cambridge University Press, 1984.
23. Friese, M. A., & Banerjee, S. (1995). FT-IR spectroscopy. In T. E. Conners & S. Banerjee (Eds.), Surface analysis of paper (pp. 119-141). Boca Raton, FL: CRC Press.
24. Garoufallou, E., Banos, V., & Koulouris, A. (2013). Solving aggregation problems of Greek cultural and educational repositories in the framework of Europeana. International Journal of Metadata, Semantics and Ontologies (IJMSO), 8(2), 134–144. doi:10.1504/IJMSO.2013.056602.
25. Haykin S. (1999). Neural Networks: a comprehensive foundation. 2nd ed., Prentice Hall, 1999.
26. Hon, D. N.-S. (1986). Fourier Transform IR spectroscopy and Electron Spectroscopy for Chemical Analysis. Use in the study of paper documents. In H. L. Needles & S. H. Zeronian (Eds.), Historic textile and paper materials. Conservation and characterization (pp. 349-361). Washington, DC: American Chemical Society.
27. Honkela T., Kaski S., Lagus K., and Kohonen T. (1996). Newsgroup exploration with WEBSOM method and browsing interface. Technical Report A32. Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland, 1996.
28. Joachims Thorsten. A probabilistic analysis of the Rocchio algorithm with TF.IDF for text categorization. In ICML-97, 1997.
29. Johansson, A., & Lennholm, H. (2000). Influences of SO2 and O3 on the ageing of paper investigated by in situ diffuse reflectance FTIR and time-resolved trace gas analysis. Applied Surface Science, 161, 163-169.
30. Kato, K. L., & Cameron, R. E. (1999). Structure-property relationships in thermally aged cellulose fibers and paper. Journal of Applied Polymer Science, 74(6), 1465-1477.
31. Kohonen T. (1989). Self-organization and associative memory. 3rd ed. Berlin: Springer-Verlag, 1989.
32. Kohonen T. (1998). Self-organization of very large document collections: state of the art. In Niklasson, L., Boden, M., and Ziemke, T., editors, Proceedings of ICANN98, the 8th International Conference on Artificial Neural Networks, Vol. 1, pages 65-74, London, 1998.
33. Kohonen T., Kaski S., Lagus K., Salojärvi J., Honkela J., Paatero V., and Saarela A. (2000). Self-organization of a massive document collection. In IEEE Transactions on Neural Networks, Vol. 11, No. 3, May 2000.
34. Koninklijke Bibliotheek (2009). Europeana, available at: http://www.europeana.eu (31 July 2014).
35. Kotsiantis S.B., Zaharakis I.D., Pintelas P.E. (2006). "Machine learning: a review of classification and combining techniques". Artificial Intelligence Review 26 (3), 159-190.
36. Kupiec J., Pedersen J., Chen F. (1995). A Trainable Document Summarizer. In Proceedings of the 18th ACM-SIGIR Conference, 88-97.
37. Lacy Lee W. (2005). "Chapter 10". OWL: Representing Information Using the Web Ontology Language. Victoria, BC: Trafford Publishing.
38. Lewis D.D., Schapire R., Callan J.P. and Papka R. Training algorithms for linear text classifiers. In Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996.
39. Lin X., Soergel D., Marchionini G. (1991). A self-organizing semantic map for information retrieval. In Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1991.
40. Littlestone N.(1988). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Machine Learning, 2:285-318, 1988.
41. Liu Ling. (1999). "Query Routing in Large-scale Digital Library Systems", Data Engineering, 1999 Proceedings, 15th International conference on Sydney, NSW, Australia, Mar. 23-26, 1999, Los Alamitos, CA, USA, IEEE Computer Society, Mar. 23, 19999, pp. 154-163.

42. Luhn H. (1958). The Automatic Creation of Literature Abstracts. In IBM Journal of Research and Development 2(2):159-165.
43. Margutti S., Conio G., Calvini P., & Pedemonte E. (2001). Hydrolytic and oxidative degradation of paper. Restaurator, 22(2), 67-83.
44. McCallum A. and Nigam K. (1998). A comparison of even models for naïve bayes text classification. In Workshop on Learning for Text Categorization, AAAI, 1998.
45. Mitchell Tom M. (1997). Machine Learning. Mc Graw Hill, International Editions, 1997.
46. Moropoulou A., & Zervos S. (2003). The immediate impact of aqueous treatments on the strength of paper. Restaurator, 24(3), 160-177.
47. Myaeng Sung Hyon and Jang Dong-Hyun. (1997). Development and Evaluation of a Statistically-based Document Summarization System. In Advances in Automatic Text Summarization, Ed. I. Mani and M. Maybury, Cambridge MA: MIT Press, 1998, 61-70.
48. National Documentation Centre (2010). National Archive of PhD Theses, available at: http://phdtheses.ekt.gr/eadd/?locale=en (31 July 2014).
49. Pavia D. L., Lampman G. M., & Kriz G. S. (1996). Introduction to spectroscopy (2nd Edition ed.). Fort Worth, TX: Saunders College Publising.
50. Proniewicz L.M., Paluszkiewicz C., Weselucha-Birczynska A., Majcherczyk H., Baranski A., & Konieczna A. (2001). FT-IR and FT-Raman study of hydrothermally degradated cellulose. Journal of Molecular Structure, 596, 163-169.
51. Roberts J. C. (1996). The chemistry of paper. Cambridge, UK: The Royal Society of Chemistry.
52. Santos S. M., Carbajo J. M., Quintana E., Ibarra D., Gomez N., Ladero M., . . . Villar J. C. (2014). Characterization of purified bacterial cellulose focused on its use on paper restoration. Carbohydrate Polymers(0), in press. doi: http://dx.doi.org/10.1016/j.carbpol.2014.03.064
53. Scholkopf C., Burges J.C., Smola A.J. (1999) Advances in Kernel Methods. MIT Press.
54. Sistach, M. C., Ferrer, N., & Romero, M. T. (1998). Fourier Transform Infrared Spectroscopy applied to the analysis of ancient manuscripts. Restaurator, 19(4), 173-186.
55. Sjöström, J., & Brolin, A. (1996). Bleached pulp composition and Its determination. In C. Dence & D. Reeve (Eds.), Pulp bleaching, principles and practice (pp. 677-693). Atlanta, GA: Tappi Press.
56. Soares, S., Ricardo, N. M. P. S., Jones, S., & Heatley, F. (2001). High temperature thermal degradation of cellulose in air studied using FTIR and 1 H and 13 C solid-state NMR. European Polymer Journal, 37, 737-745.
57. Sobucki, W., & Drewniewska-Idziak, B. (2003). Survey of the preservation status of the 19th and 20th century collections at the National Library in Warsaw. Restaurator, 24(3), 189-201.
58. Stefanis, E., & Panayiotou, C. (2010). Deacidification of Documents Containing Iron Gall Ink with Dispersions of Ca (OH) 2 and Mg (OH) 2 Nanoparticles. Restaurator, 31(1), 19-40.
59. Triantafyllou I., Demiros I., Piperidis S. (2001). "Two Level Self-Organizing Approach to Text Classification", RANLP-2001: Recent Advances in NLP.
60. Triantafyllou Ioannis, Carayannis George. (2003). "Architectures and Techniques for Monolingual and Multilingual Information Retrieval Systems in a SOM Framework". WSEAS Transactions on Systems, Issue 3, Volume 2, July 2003, pages 589-597.
61. Webb G.I., Boughton J., Zheng F., Ting K.M., and Salem H. (2012). Learning by extrapolation from marginal to full-multivariate probability distributions: Decreasingly naive Bayesian classification. Machine Learning. 86(2): 233-272. Netherlands: Springer.
62. Yang Yiming and Pederson Jan. (1997). Feature selection in statistical learning of text categorization. In ICML-97, 1997.
63. Yang, C. Q., Freeman, J.M. (1991). Photo-oxidation of cotton cellulose studied by FTIR photoacoustic spectroscopy. Applied Spectroscopy, 45(10), 1695-1698.
64. Zervos, S. (2007). Evaluating treatments of paper using statistically valid test methods. Part II: Experimental setup and protocol. Restaurator, 28(4), 256-288.
65. Zervos, S. (2010). Natural and accelerated ageing of cellulose and paper: A literature review. In A. Lejeune & T. Deprez (Eds.), Cellulose: Structure and Properties, Derivatives and Industrial Uses (pp. 155-203). New York: Nova Publishing.
66. Zervos, S., & Moropoulou, A. (2006). Methodology and criteria for the evaluation of paper conservation interventions. Literature review. Restaurator, 27(4), 219-274.