8th IEEE International Symposium on Applied Computational Intelligence and Informatics • May 23–25, 2013 • Timisoara, Romania

Hybrid approach for visualization of documents clusters using GHSOM and Sammon projection

P. Butka[*], J. Pócsová[**]

[*] Technical University of Košice, Faculty of Electrical Engineering and Informatics, Department of Cybernetics and Artificial Intelligence, Letná 9, 040 01, Košice, Slovakia

[**] Technical University of Košice, BERG Faculty, Institute of Control and Informatization of Production Processes, Kosice, Slovakia

E-mails: peter.butka@tuke.sk, jana.pocsova@tuke.sk

Abstract—**This paper presents the hybrid approach for visualization of documents sets by the combination of hierarchical clustering method, based on the Growing Hierarchical Self-Organizing Maps algorithm, and Sammon projection. Algorithms based on the self-organizing maps provide robust clustering method suitable for visualization of larger number of documents into the grid-based 2D maps. Sammon projection is nonlinear projection method suitable mostly to visualization of smaller sets of object on (usually 2D) maps based on the projections. Here we have implemented and tested combination of these approaches, where starting set of documents is organized using GHSOM to subsets of similar documents, then for clusters at the end of clustering phase, with smaller number of inputs, Sammon maps are created in order to provide distinction also for documents in these clusters. The method for extraction of characteristic terms based on the information gain analysis was used for description of clusters. Existing library JBOWL was used for implementation of the hybrid algorithm. For testing purposes, the documents in English language were used.**

I. Introduction

Currently, there are several systems for clustering and visualisation of textual data. Many of them are based on the self-organizing clustering methods. The main (and basic) algorithm in this are is known as Self-Organizing Map (or SOM) and it was introduced and described by Kohonen (therefore it is also known as Kohonen map). Extensive introduction and description of its principles can be found in [1]. The basic feature of models based on the SOM architecture is its topologically preserved mapping, i.e., vectors representing data objects from input space (usually high dimensional, especially for textual data) are mapped onto output space (usually two dimensional map), but if two vectors were close (topologically) to each other in original space, their projected vectors will stay close in topological manner also in output space (they will be close to each other also on 2D map).

For some applications, the problem with classical architecture of SOM (Kohonen map) is in its fixed structure (the map has fixed size). Several models were designed in order to make the structure more adaptive, but with the grid structure behind (for easier orientation on the map). The model similar to the classical SOM, Growing Grid (designed by Fritzke [2]), is actually the SOM structure, which is capable to grow dynamically during the learning iterations by addition of rows or columns of new clusters (neurons). Other adaptive method is based on the application of hierarchical decomposition approach, i.e., the hierarchical structure of independent maps is created, where to every map element (cluster) there is possibility to add new sub-map ("children" map for "parent" cluster) for better distinction of inputs from the parent cluster. This architecture is called Hierarchical Feature Map (HFM, [3]). These two approaches can be combined into one adaptive model. Algorithm GHSOM (Growing Hierarchical SOM) combines previous models [4], i.e., every layer of its hierarchical structure consists of set of independent maps, which are able to adapt their sizes according to the needs of input vectors of data related to the particular map (and its clusters).

Sammon projection (or Sammon's mapping) [5] is well-known nonlinear projection method, mostly suitable for smaller sets of input objects. The main goal of the projection is to model (as precisely as possible) real distances of objects (in original) high-dimensional space by distances in projected (usually) two dimensional space (with maximum effort to preserve relative distances between objects). The main advantage is visualisation effect of the projection, especially if number of objects is low. With the higher number of objects (and especially also for high number of attributes) the modelling of distances is quite problematic, i.e., visualisation of large documents sets using Sammon projection is not very suitable (both from the point of usability and computing times).

In this paper we provide hybrid approach to visualisation of documents set based on the combination of hierarchical self-organizing maps (GHSOM) and Sammon projection. While self-organizing maps introduce robust clustering method suitable for organization of larger collections of textual documents in grid-based structure (map consists of clusters organized in two dimensional grid, e.g. 3x3, 4x5, etc.) of clusters (hierarchically organized using top-down approach), Sammon projection provides a method for visualisation of particular documents (represented by their vectors) as independent points in two dimensional space with the same goal (as in SOM) to preserve the topology according to the distances between vectors (of course, just as projection, not according to some centroids), but with the real distances (not on some predefined grid) on projected map. As GHSOM is more suitable for larger collection and do not recognize particular documents in its smallest clusters, Sammon projection (usually good for smaller

978-1-4673-6400-3/13/$31.00 ©2013 IEEE

sets) can be used to show differences between particular vectors in these 'leaf' clusters (those clusters in GHSOM, which are not clustered further in next layer - sub-map),

We have tested the combination of mentioned approaches on selected set of textual documents (tested documents were in English). Input collection was first clustered using GHSOM algorithm in order to find smaller subsets of similar documents. Then for 'leaf' clusters Sammon map is produced using the computed projection. For description of clusters, method for extraction of characteristic terms based on the information gain analysis was used. For the implementation needs, our JBOWL library for processing of text documents was used [6].

In next section, general approach and particular methods are described. In following section we have described the common integration and implementation of system is described, together with the short description of experiments.

II. Applied Approaches and Methods

A. Preprocessing of documents and their representation

For the needs of the presented algorithms it is necessary to preprocess input data set and produce document-term matrix of weights (so-called vector-based representation of documents set), where each row represents one document using its vector (with words or terms as attributes). Preprocessing consists of following steps (in our case):

1. Tokenization
2. Stop-words elimination
3. Stemming
4. Term selection filtering

Tokenization transforms the input text into tokens. It is necessary step of the preprocessing, other steps are optional, but are used in order to reduce number of terms (attributes) in the final vector-based representation. A set of tokens represents the basic vocabulary of processed dataset. In next step, stopwords (meaningless words like 'and', 'of', etc.) are removed from the vocabulary (using the comparison with list of stopwords, also known as stop-list, which is available for English language). In next step, tokens (which were not removed) are transformed on their stems (root form, e.g., 'wins' and 'win' are transformed to one root - 'win'). This again leads to the reduction of vocabulary (for different morphological forms we will get only root form). The last step is based on the selection of terms according to their occurrence frequencies. If some term is very usual within the dataset or is only in low number of documents, such term is also removed from the vocabulary. Therefore, it is possible to setup two thresholds for minimal and maximal possible frequency of term within the documents set. The input collection of the documents is then represented by the document-term matrix with the values based on the selected weighting scheme. For this issue we have used well-known TF-IDF scheme, which assigns the weight to every pair document-term using this formula

$$w_{ij} = tf_{ij} \,.\, \log\left(\frac{ndocs}{df_j}\right)$$

where tf_{ij} is the number of occurrences of term j in document i, $ndocs$ is number of documents in the collection and df_j is the number of documents in which term j is used (inverse document frequency). The result of the process is that we have document-term matrix with weights w_{ij} characterizing the documents set in the form called vector space representation.

B. Algorithm GHSOM

Here, we will shortly describe the algorithm which is used for clustering of the collection of documents in order to find smaller subsets of similar documents on different levels of hierarchy of maps. At the start, starting version of top-level map is initialised (with $m \times n$ clusters). Then, input vectors are presented to the input of network. For every cluster its activation to particular document is computed. Cluster (formally 'neuron') with the highest activation (if we will use for example cosine similarity, as it was in our experiments) is winner. The vector of winner together with the vectors of its neighbourhood clusters within the grid structure (but with the smaller difference) is adapted in order to become more similar to input vector (using next two formulas):

$$INF = 1 - \left(\frac{dist}{NGH + 0.5}\right),$$

$$c_j^* = c_j + LR \,.\, INF \,.\left(v_j - c_j\right),$$

where $dist$ is distance between winner (cluster) and updated cluster c, NGH is neighbourhood parameter (it describes the radius of neighboured clusters used for adaption). Then, INF is computed ratio of influence of current input v to updated weights of cluster c, LR is learning factor (decreasing during the iterations), c^* means the new value of vector for cluster c.

First iteration process (one iteration = every document from the collection is used once as input vector) ends with the achieved (predefined) maximum number of iterations. Then the variability (or mean quantization error - MQE) of every neuron is computed, i.e., it is variability of documents asserted to the particular cluster computed as averaged distance of documents within cluster to its centroid. Also, MQE of the whole map is computed, i.e., it is averaged variability (MQE) of clusters on the map. After these steps MQE of map is tested for condition:

$$MQE \geq \tau_1 .mqe_0 ,$$

where MQE is variability of the map and mqe0 is variability of the data itself (variability of all input vectors according to their average vector computed through all data). If this condition is true, then it is still necessary to add neurons. Therefore, parameter τ_1 represents threshold for addition of clusters to the particular map. Before the addition, we need to find most variable cluster ('error neuron') and its most distanced neighbour (according to the metric used in original input space). Then whole block of clusters is added between them (complete row or column, depends on their relative positions on the map). Then map is relearned. If variability of map will decrease and aforementioned condition is not valid, growing of the map is finished. After the growing phase, every cluster is tested for possibility to expand on new level of hierarchy (sub-map for data from that cluster). So if the condition

$$mqe \geq \tau_2 .mqe_0$$

is fulfilled, current cluster is expanded on new level of hierarchy as a new sub-map (mqe is error of currently analysed cluster). The condition represents relation between variability of particular cluster and mpq_0. Parameter τ_2 is threshold for expansion of clusters. New sub-map starts with 2x2 grid of clusters. Every created sub-map learns in the same way with possibility to grow up and (if it is necessary) to expand some of its clusters to new sub-maps. In our case, we have added another parameter to the condition for expansion of clusters – minimal number of documents in cluster for expansion (in order to avoid large division to very small subsets). Algorithm GHSOM works in top-down way and divides the dataset of input vectors to smaller subsets organized in hierarchy of maps. The algorithm finishes when there is not suitable cluster (in any map) for expansion.

C. Sammon projection

The main goal of Sammon projection [5] is transformation, which minimizes defined error function for projection of distances between vectors of original (high-dimensional) space. This minimization minimizes the error using assignment of points in lower dimension in the way for which the distance between points is as similar as possible to their distance in higher dimension. The simplest approach is based on the gradient iteration method for optimization.

Let n objects are represented by the points in t-dimensional space. The goal of the Sammon projection is to find n points in g-dimensional space (where $d < g$) for which the corresponding transformed distances are in best approximation with original distances. Let for two points of input set with indexes i and j (i,j=1,…, n), d^*_{ij} is their distance in original g-dimensional space and d_{ij} is their distance in projected d-dimensional space. Then quality of projection is defined according to this error function (called ‘stress function’):

$$E = \frac{1}{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} d^*_{ij}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{\left(d^*_{ij} - d_{ij}\right)^2}{d^*_{ij}}$$

The values of E is from <0,1> interval, where 0 indicates lossless projection (mapping). Next, we will assume only projection to two dimensional space (g = 2). The optimization problem for projection for higher m is complex. Stress function is optimal when original and projected distances are equivalent. In practice, there is always some distortion (and it is higher with the increasing value of E). For finding the projection map, we have to start with the initialization points, then E is computed and projected points are iteratively adapted in order to improve stress function. Iterations end when minimum of E is found. Let $E(m)$ is stress function in iteration m, for which

$$E(m) = \frac{1}{\sum_{i<j}^{n} d^*_{ij}} \sum_{i<j}^{n} \frac{\left[d^*_{ij} - d_{ij}(m)\right]^2}{d^*_{ij}},$$

where $d_{ij}(m) = \sqrt{\sum_{k=1}^{d} \left(y_{ik}(m) - y_{jk}(m)\right)^2}$.

Now, the formula for calculation of $m+1$iteration is:

$$y_{pq}(m+1) = y_{pq}(m) - (MF) * \Delta_{pq}(m),$$

where

$$\Delta_{pq}(m) = \frac{\partial E(m)}{\partial y_{pq}(m)} / \left|\frac{\partial^2 E(m)}{\partial y_{pq}(m)^2}\right|$$

and for correction factor MF (also called 'magic factor') is usually used the value from interval <0.3,0.4>. Partial derivations are computed using next two formulas :

$$\frac{\partial E(m)}{\partial y_{pq}} = -\frac{2}{c}\sum_{\substack{i=1\\ j\neq p}}^{N} \frac{(d^*_{pj} - d_{pj}(m))}{d_{pj}(m)\cdot d^*_{pj}}\left(y_{pq}(m) - y_{jq}(m)\right)$$

$$\frac{\partial^2 E(m)}{\partial y_{pq}{}^2} = -\frac{2}{c}\sum_{\substack{j=1\\ j\neq p}}^{N} \frac{1}{d^*_{pj}(m)\cdot d_{pj}} \cdot \left[\left(d^*_{pj} - d_{pj}(m)\right) - \left(\frac{\left(y_{pq}(m) - y_{jq}(m)\right)^2}{d_{pj}(m)}\right)\left(1 + \frac{d^*_{pj} - d_{pj}(m)}{d_{pj}(m)}\right)\right]$$

where $c = \sum_{i<j}^{n} d^*_{ij}$.

Detailed proofs of equations can be found in [7].

D. Extraction of characteristic terms

For description of particular clusters it is possible to use several methods for extraction of characteristic terms (words or keywords of particular subset of documents). One of the options is to use approach based on the information gain evaluation of terms. This is often used as a criterion for assignment of correct keyword. It rates how much information is included in particular term in order to predict class defined by the occurrence or absence of term in some document (or document set).

Let we have h categories in target space and i-th category is defined as C_i. Information gain of term t is defined as:

$$G(t) = -\sum_{i=1}^{h}\Pr(C_i)\log\Pr(C_i) + \Pr(t)\sum_{i=1}^{h}\Pr(C_i \mid t)\log\Pr(C_i \mid t) +$$
$$+ \Pr(\bar{t})\sum_{i=1}^{h}\Pr(C_i \mid \bar{t})\log\Pr(C_i \mid \bar{t})$$

where $\Pr(X) = \frac{N_x}{N}$ is probability of category X, conditional probability of category X according to the occurrence of term t is $\Pr(X \mid t) = \frac{N_{tX}}{N_t}$ and $\Pr(X \mid \bar{t}) = \frac{N_{\bar{t}X}}{N_{\bar{t}}}$ represents conditional probability of category X according to the absence of term t. Number N represents number of documents covering the particular condition (e.g. N_t is number of documents which contain term t). Information gain is computed for every term against cluster (which is used as assigned category) and all terms with lower information gain than predefined threshold are removed from the description of current cluster, or just selected number of highly ranked terms is used for every cluster.

III. Hybridization of Approaches and Experiments

A. Implementation of hybrid approach

The combination of GHSOM algorithm was implemented (in Java) using the library created specially for processing and text-mining of text documents set (see [6] for reference). Our process for creation of the output hybrid model can be described as follows:

1. Preprocessing
2. Creation of combined hybrid model

3. Model visualisation

The preprocessing phase follows the step defined in sub-section II.A, where for stemming step simple Porter stemmer was used [8]. It is simple algorithm based on the pruning of suffixes, but for English text is quite successful method.

Creation of model is realized by the integration of Sammon projection into the GHSOM model built using JBOWL. Actually, the GSOM (Growing Grid) algorithm was implemented in JBOWL as independent algorithm. Then GHSOM is implemented as HierarchicalAlgorithm class extension, which uses GSOM as its base algorithm for particular layers. Thanks to that GHSOM implementation is able to expand to new sub-maps easily, all is processed in the form of tree organized top-down, i.e., from basic GSOM model (first map) down through the layers to 'leaf' nodes, which are maps without expanded clusters ('leaf' GSOM models).

According to the relations between documents hierarchy of them is created using GHSOM. Now, thanks to the implementation of the algorithm in JBOWL, it is easy to get unexpanded nodes (clusters) and create Sammon map (using implemented SammonAlgorithm class). Practically, for 'leaf' nodes it is possible to run some LeafAlgorithm (here is SammonAlgorithm used for our case), which supports the realization of other algorithm in the moment when expansion using new GSOM sub-maps is finished for the cluster (this is covered by the stop condition for cluster expansion mentioned in sub-section II.B).

SammonAlgorithm is also centroid-based model, but its values are directly coordinates of projected documents in two dimensional space. At the start, the algorithm reads the number of input vectors (only those assigned to the particular cluster are used for Sammon projection computation). Then vectors of centroids are created (two-dimensional representation of documents) and randomly initialized. In next step distances between input vectors (distanceMatrix) and projected vectors (distanceSammon) are computed, together with the stress function E. The iteration process follows with the goal to minimize stress function using the method described in sub-section II.C.

The result of the whole building model step is created hybrid hierarchical serialized model, which consists of ordered and organized models of maps created by the GSHOM and Sammon projection. This model is then input of the visualisation step. If we have used only GHSOM algorithm, the result of visualisation step is the set of HTML pages, where each page is one GSOM map in the hierarchy. Every map is realized as table, where every cell represents one cluster described by its position on the map (row and column), number of assigned vector and the set of characteristic terms for this cluster (extracted using the method based on the information gain analysis described in sub-section II.D).

Extension element in our case is the existence of other HTML pages, which shows the Sammon maps at the end of the hierarchies, i.e., leaf nodes of the hierarchies. In our case HTML code for cluster is extended with the link to subpage containing the Sammon map of such leaf cluster. Therefore, the set of HTML pages of GHSOM maps and Sammon maps, connected through necessary links, are the result of visualisation part of the process. On the page with Sammon map is link to 'parent' cluster as well as visualised documents, which are presented according to the coordinates from the output of centroids vectors. The values of coordinates are normalized to (-1,1). Using the mouse movement over the documents, it is possible to show also characteristic terms for particular documents on the map. Created set of pages provides hierarchical structure, which can be simply browsed.

B. Illustrative example from experiments

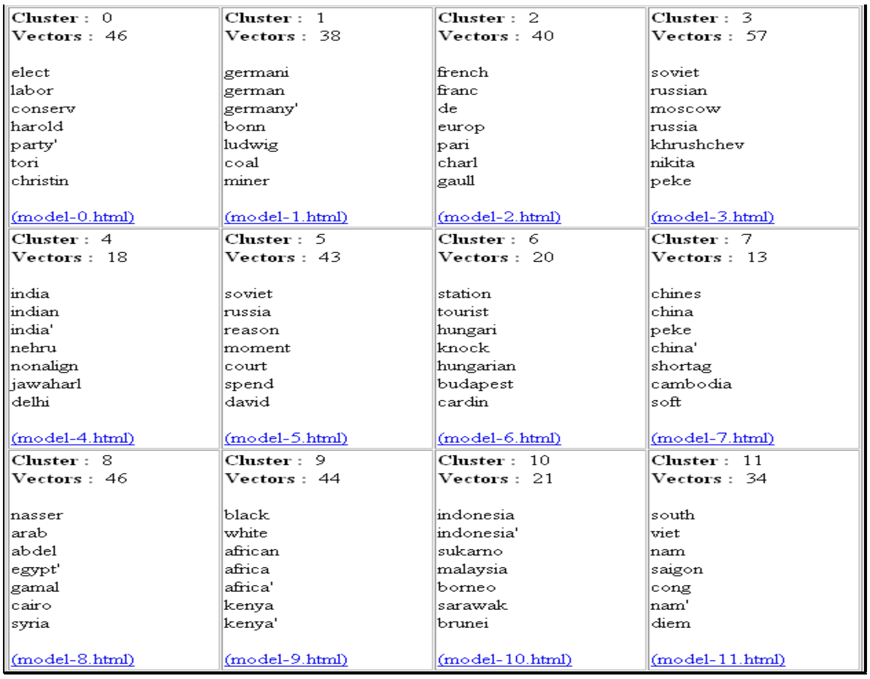
For our experiments, dataset containing the news articles from Times newspaper from 60's was used. It is relatively small dataset with 420 documents on different topics like international relationships, economic, political situation and history of different countries and regions, Vietnam War, etc. We have used this dataset due to fact that it is not too large for first testing of our hybrid method and it is well-known to us from our previous work. Another dataset used for the testing was standard collection known as Reuters-21578, also containing the articles from the newspapers, with 7769 training documents.

After preprocessing phase, number of terms was influenced by the parameters for minimal and maximal frequency of terms occurrences, e.g., for presented illustrative example (based on Times dataset) minimum was setup to 5 documents and maximum to 100 documents. Then number of attributes (terms) was 1925. Example of the part of the map from GHSOM can be found in Fig. 1 (it is first level map for Times collection).

As we can see, GHSOM algorithm divides the set of documents to clusters of similar documents, i.e., it produces categories of them. Several facts are known about clusters - its ID on map, number of assigned vectors, some characteristic terms (in this example 7 was used as number of best terms after information gain analysis to be shown for clusters), link to sub-map (here can be link to expanded GSOM or to Sammon map, if it is a 'leaf' node). From the description of clusters it is possible to estimate most probable topics of documents in clusters and follow the links in order to browse the structure of pages.

Sometimes, the clustering is not able to separate documents exactly to one cluster and it is needed to make some more detailed approach. Here the Sammon projection can be helpful. Example of the Sammon projection for one selected cluster is presented in Fig.2. As we can see, now it is possible to separate also documents within one cluster. Also, due to previous clustering by GHSOM, Sammon map is simpler and quite comprehensive thanks to smaller group of documents to be shown on the map.

Implementation of Sammon projection into the hybrid algorithm with GHSOM the created system is able to prepare 2D based visualisation of documents sets in form of HTML pages. Extracted keyword in Sammon maps provided more detailed view of visualised documents. Sammon mapping was able to visualise the similar documents quite accurately. The description based on the keywords extraction was also helpful as evaluation for correctness of projections, i.e., clusters contain documents with similar topics and documents shown near to each other on the map were more similar than documents shown on the other side maps. In the future we would like to implement this system into the web portal, where



Cluster : 0 Vectors : 46 elect labor conserv harold party' tori christin (model-0.html)	Cluster : 1 Vectors : 38 germani german germany' bonn ludwig coal miner (model-1.html)	Cluster : 2 Vectors : 40 french franc de europ pari charl gaull (model-2.html)	Cluster : 3 Vectors : 57 soviet russian moscow russia khrushchev nikita peke (model-3.html)
Cluster : 4 Vectors : 18 india indian india' nehru nonalign jawaharl delhi (model-4.html)	Cluster : 5 Vectors : 43 soviet russia reason moment court spend david (model-5.html)	Cluster : 6 Vectors : 20 station tourist hungari knock hungarian budapest cardin (model-6.html)	Cluster : 7 Vectors : 13 chines china peke china' shortag cambodia soft (model-7.html)
Cluster : 8 Vectors : 46 nasser arab abdel egypt' gamal cairo syria (model-8.html)	Cluster : 9 Vectors : 44 black white african africa africa' kenya kenya' (model-9.html)	Cluster : 10 Vectors : 21 indonesia indonesia' sukarno malaysia borneo sarawak brunei (model-10.html)	Cluster : 11 Vectors : 34 south viet nam saigon cong nam' diem (model-11.html)

Figure 1. First level map of GHSOM result for Times60 colection. It is possible to find most probable topics (mostly differed by regions at this level) from their characteristic terms, e.g., topics related to the France, situation in Vietnam in 60's, problems in Indonesia and near region, or articles related to India, Germany, Soviet union (and its influence), etc. Links at the bottom of clusters descriptions lead to their sub-maps (another GSOM or Sammon).

cloud-based implementation based on GridGain software (using similar techniques to [9]) will be used in order to achieve scalable solution for visualisation of larger sets of documents. Also, we would like to implement support for inclusion of our algorithm into text-mining workflows [10], which will be semantically annotated and defined for usage in different hybrid text-mining tasks [11].

Due to fact that GHSOM and Sammon can be easily applied also in other domains, we would like to provide visualisation for data from other data mining tasks, especially analysis of logs or prediction tasks (see [12] and [13] for the reference of potential data mining tasks). The universality of the approach is straightforward, but (of course) some of the steps will be applied differently. Also we would like to apply similar hybrid approach, which combines Formal Concept Analysis with GHSOM. Some effort was already done in [14], in order to provide information retrieval method, based on the generalized one-sided concept lattices (theoretically introduced by [15] and [16]).

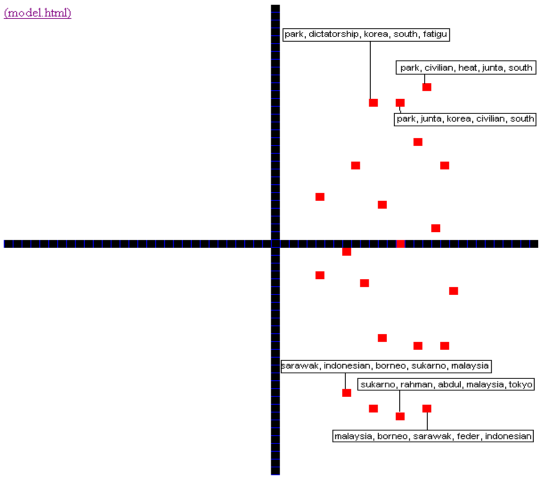


Figure 2. Example of Sammon map at the bottom of the structure. Here we can see 'leaf' node with 21 documents. Usage of mouse leads to alternative text shown for every document (extracted keywords), we have extracted some of them and add them to this figure in order to have them shown together. As we can see, in the upper part of the map there is one group of documents (described by the words like korea, park, dictatorship, etc.), while at the bottom of map similar problems are described, but for different region (terms like indonesia, borneo, malaysia, etc.).

IV. Conclusion

In this work we have presented relatively simple solution for hybrid visualisation methods which combines robust SOM-based hierarchical clustering approach (known as GHSOM algorithm) and Sammon projection. Our implementation and testing on selected datasets proved that while SOM-like solution is better in organization of larger datasets, but for smaller subsets of documents Sammon projection is suitable to distinct also individual documents. The advantages of both approaches were combined and hybrid solution was implemented using our text-mining library JBOWL. Therefore, at the end of the expansion of clusters, it is now possible to show Sammon map of cluster in order to also understand differences between documents within particular clusters. In the future we would like to make test on larger datasets and prepare more scalable solution, which can be used as visualisation portal and will be realized on some cloud-based technology (like GridGain platform).

Acknowledgment

This work was supported by the Development of the Center of Information and Communication Technologies for Knowledge Systems (ITMS project code: 26220120030)(25%), by the Slovak VEGA Grant No. 1/1147/12 (25%), and by the Slovak Research and

Development Agency under contracts APVV-0208-10 (25%) and APVV-0482-11 (25%).

References

[1] T. Kohonen, *Self-Organizing Maps*. Third edition. Springer Series in Information Sciences 30, Springer, 2001.

[2] F. Fritzke, “Growing Grid - a self-organizing network with constant neighbourhood and adaptive strength”, *Neural Processing Letters*, vol. 2, no. 5, pp. 9-13, 1995.

[3] D. Merkl, “Explorations of text collections with Hierarchical Feature Maps”, in *Proceedings of International ACM SIGIR Conference in Information Retrieval*, Philadelphia, pp. 186-195, 1997.

[4] M. Dittenbach, D. Merkl, A. Rauber, “Using growing hierarchical self-organizing maps for document classification”, in *Proceedings of ESANN*, Bruges, Belgium, pp. 7-12, 2000.

[5] J.R. Sammon, “A nonlinear mapping for data structure analysis”, *IEEE Transactions on Computers C*, vol. 18, pp. 401-409, 1969.

[6] P. Bednar, P. Butka, J. Paralic, “Java Library for Support of Text Mining and Retrieval”, in *Proceedings of Znalosti 2005*, Stara Lesna, Slovakia, pp.162-169, 2005.

[7] C.S. Tsai, “Visual Display Techniques”, Dissertation Thesis, Department of Computer Science and Information Engineering, National Chi-Nan University, 2003.

[8] M.F. Porter, “An algorithm for suffix stripping”, In *V Program*, vol. 14, no. 3, pp. 130-137, 1980.

[9] M. Sarnovsky, T. Kacur, “Cloud-based classification of text documents using the Gridgain platform”, in *Proceedings of the 7th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2012)*, Timişoara, Romania, pp. 241-245, 2012.

[10] M. Sarnovsky, M. Paralic, “Text mining workflows construction with support of ontologies”, in *Proceedings of SAMI 2008*, Herlany, Slovakia, pp. 173-177, 2008.

[11] M. Babik, M. Sarnovsky, Z. Durcik, “Establishing semantic annotation and execution of the text-mining services”, in *Proceedings of WIKT 2008*, Smolenice, Slovakia, pp. 78-82, 2009.

[12] J. Paralic, C. Richter, F. Babic, J. Wagner, M. Racek, “Mirroring of knowledge practices based on user-defined patterns”, *Journal of Universal Computer Science*, vol. 17, no. 10, pp. 1474-1491, 2011.

[13] F. Babic, P. Bednar, F. Albert, J. Paralic, J. Bartok, L. Hluchy, “Meteorological Phenomena Forecast Using Data Mining Prediction Methods”, In: P. Jedrzejowicz, N.T. Nguyen, K. Hoang (eds.) *ICCCI 2011*, Part I, LNCS, vol. 6922, pp. 458-467, 2011.

[14] P. Butka, J. Pócsová, J. Pócs, “A Proposal of the Information Retrieval System based on the Generalized One-Sided Concept Lattices", In: *Applied Computational Intelligence in Engineering and Information Technology*, TIE (series: Topics in Intelligent Engineering and Informatics), vol. 1, Springer Verlag, pp. 59-70, 2012.

[15] J. Pocs, “Note on generating fuzzy concept lattices via Galois connections”, *Information Sciences*, vol. 185, no. 1, pp. 128–136, 2012.

[16] J. Pocs, “On possible generalization of fuzzy concept lattices using dually isomorphic retracts”, *Information Sciences*, vol. 210 , pp. 89-98, 2012.