Discovery of Hierarchical Thematic Structure in Text Collections with Adaptive Resonance Theory

Louis Massey

Royal Military College

Department of Mathematics and Computer Science

Kingston, Ontario, Canada, K7K 7B4

1-613-541-6450

1-613-541-6584

massey@rmc.ca

This research was supported in part by the National Defense Academic Research Program (ARP) under grant 743321.

Abstract This paper investigates the abilities of Adaptive Resonance Theory (ART) neural networks as miners of hierarchical thematic structure in text collections. We present experimental results with binary ART1 on the benchmark Reuter-21578 corpus. Using both quantitative evaluation with the standard F_1 measure and qualitative visualization of the hierarchy obtained with ART, we discuss how useful ART built hierarchies would be to a user intending to use it as a means to find and access textual information. Our F_1 results show that ART1 produces hierarchical clustering that exhibit a quality exceeding k-means and a hierarchical clustering algorithm. However, we identify several critical problem areas that would make it rather impractical to actually use such a hierarchy in a real-life environment. These predicaments point to the importance of semantic feature selection. Our main contribution is to test in details the applicability of ART to the important domain of hierarchical document clustering, an application of Adaptive Resonance that had received little attention until now.

Topics hierarchy, Hierarchical Clustering, Adaptive Resonance Theory, ART, Text Mining

Introduction

There has been a keen interest in recent years to automate the construction of thematic hierarchical structures for document collections. Supervised techniques [1], [2] as well as unsupervised clustering [3]-[5] have been investigated. Organizing document collections in a hierarchical manner has always been deemed an important endeavor as evidenced by the Dewey classification system used by libraries and by the hierarchical topic structure used by the National

Health Institute for MEDLINE. A topics hierarchy allows easier access to information by partitioning the search space a user needs to consider. Indeed, the user can focus her search on the most promising branch of the topics tree, drilling down her way from more general to more specific. The automation of thematic hierarchies construction is thus becoming increasingly important given the growth of vast electronic text collections, including the Internet. Automation for instance would allow much better coverage and more efficient construction of the web directories currently found as additional offerings on web search engines sites. Automated topics hierarchy generation can also help with the organization of very large text corpus [6] used by industry, government and science knowledge workers. Another application of hierarchical clustering is to facilitate information retrieval by, for instance, organizing results returned by a search engine in a tree structure [7].

A full review of existing neural network based methods for document clustering and organization is given in [8]. A promising yet little studied approach to unsupervised hierarchical text clustering is the Adaptive Resonance Theory (ART) [9]. ART vigilance parameter is particularly well suited to discover hierarchical structures in text, yet as far as we know only two preliminary attempts have been made to apply ART to this important problem [10], [11]. It would therefore be important to fully characterize the possibilities of ART in this area. This paper tackles this task, investigating both quantitatively and qualitatively the potential of ART as a hierarchical topic miner. This work is a continuation of an extensive program to evaluate the potential of ART in text clustering [12], [13]. Previous work was concerned with hard, flat (partitioning) clustering, while here we are dealing with hard, hierarchical clustering.

Therefore, our contribution is to test in details the applicability of ART to the important domain of hierarchical document clustering, an application of Adaptive Resonance that had received little attention until now. This work will be useful to the research community by providing a good picture of the application of this type of neural network to the task of documents hierarchical clustering. We do not aim at presenting a new neural architecture but rather at measuring the usefulness of ART networks applied to hierarchical text clustering.

The paper is organized as follows: In part II, we describe Adaptive Resonance Theory neural networks in general while in part III previous work in hierarchical clustering with ART networks is covered. Section IV presents the experimental setting and part V the experimental results and their analysis and discussion. Finally, in section VI we conclude and present some

possible future work.

Adaptive Resonance Theory

Adaptive Resonance Theory (ART) describes a type of competitive neural network. The theory was developed by Grossberg in the 70's with the objective of modeling learning systems that are both plastic and stable. Several different types of networks and implementations have since been proposed in the literature. In this paper we focus on a binary version following the implementation of Beale and Jackson [14]. The architecture of an ART1 network is summarized on Fig. 1. The network is made of two interconnected layers of neurons and of an external control system that determines the operational mode of the layers. Weights w_{ij} exist on bottom-up connections going from input neuron i to output neuron j. There is one input neuron i for each component of an input vector $\mathbf{x}_k$ of dimension N. Weights t_{ji} are attributed to top-down connections, from output neuron j to input neuron i. Each output neuron j (j=1 to M) hence has an associated vector $\mathbf{t}_j$ constituted of the weights t_{ji} from the connections out of j. A vector $\mathbf{t}_j$ corresponds to the cluster *prototype*, that is the internal representation of the category learned by neuron j. Similarly, there is an input activation vector $\mathbf{w}_j$ corresponding to the weights of connections going from the input layer to output neuron j.

The input layer (also known as F1 or comparison layer) receives inputs and propagates them on the bottom-up connections, which causes activation of neurons on the output layer (also known as the F2 or recognition layer). The dot product between input $\mathbf{x}_k$ and bottom-up connections weight vectors determines the activation u_j of each output neuron j:

$$u_j = \mathbf{x}_k \cdot w_{ij} \tag{1}$$

Competitive selection takes place between output neurons. The winner selected is the neuron j* with maximum activation j*=arg max(u_j). The cluster represented by this output neuron is deemed to be the one with the greatest correlation with the input. This constitutes a first measurement of similarity. At this point, the input signal is propagated back towards the F1 layer on the winning neuron top-down connections. Scalar product between input and weights t_{ji} takes place in transit, and when the modified signal reaches the F1 layer, the layer switches into comparison mode and a second measure of similarity is taken. This similarity is known as the *vigilance test* and is based on the vigilance parameter $\rho =]0,1]$. The test is the inequality:

$$||\mathbf{x}_k \wedge \mathbf{t}_{j*}|| \,/\, ||\mathbf{x}_k|| \;\geq\; \rho \qquad (2)$$

Following the nomenclature of Beale and Jackson [14] and of others who have presented simplified computer implemented versions of binary ART, $\mathbf{x}_k \wedge \mathbf{t}_j$ is the logical AND between a document input vector $\mathbf{x}_k$ and a prototype $\mathbf{t}_j$. $||\mathbf{x}||$ is the magnitude of vector $\mathbf{x}$ given by:

$$||\mathbf{x}|| = \sum_{i=1}^{N} x_i \qquad (3)$$

If the test is passed, the input is attributed to the winning neuron and weights are updated as such:

$$\mathbf{t}_{j*}' = \mathbf{t}_{j*} \;\wedge\; \mathbf{x}_k \quad \text{(prototype update)} \qquad (4)$$

$$\mathbf{w}_{j*}' = L\,(\mathbf{x}_k \;\wedge\; \mathbf{t}_{j*})/(L - 1 + ||\,\mathbf{x}_k \;\wedge\; \mathbf{t}_{j*}\,||\,) \qquad (5)$$

Eq. 5 might differ depending on the various implementations of ART1. L is another important parameter that affects "self-scaling" and noise sensitivity of the network with regards to superset vectors [15]). Large L values can be used to counter the effect of normalization in the update of $\mathbf{w}_j$ weights and thus favor large magnitude inputs. We set L to 2 in our experiments to avoid this situation.

On the other hand, if the vigilance test fails, the network enters search mode, considering each output neuron by decreasing activation order until one passes the vigilance test. If no neuron passes the test, a new neuron is created and the input assigned to it. Effectively, in this last case, novelty is detected and integrated, which illustrates the plasticity of ART. In any case, the network must iterate up to N-1 times through the data to achieve stabilization [16]. The prototype weight update with a logical AND guarantees a unidirectional movement of prototypes (monotonically decreasing magnitude) and thus also contributes to stability [17].

The vigilance test computes the ratio of the number of binary components common between input $\mathbf{x}_k$ and prototypes $\mathbf{t}_{j*}$ of the winning neuron vs. the number of binary components common in the input. Hence, if all active components (i.e. set to 1) in the input correspond to active components in the prototype, the ratio will be maximized and this input will surely pass the test since the maximum value allowed for ρ is 1. If on the contrary no common active component exist, the ratio will be 0 and the test will fail since $\rho > 0$ by definition.

This behavior highlights one of the advantages of ART, which is its ability to adapt to any

situation by varying the vigilance parameter. Indeed, by adjusting vigilance, the network can discover clusters of various granularities without forcing a specific number of clusters. This gives the network the unconstrained latitude to discover true structure in the data. With a high vigilance ($\rho \rightarrow 1$), the number of categories detected in the data is maximized while if $\rho \rightarrow 0$, a minimal number of clusters is discovered. In extreme cases, for $\rho = 1$, all different input objects belong to different clusters while as $\rho \rightarrow 0$ a single cluster regrouping all data objects might be created. However, in this latter case, even for a very small vigilance, the number M of clusters discovered is often greater than 1, an indication that ART vigilance permits the mining of inherent structure in data rather than arbitrarily forcing the formation of a single super-cluster. This phenomenon is known as the *minimal vigilance* [18].

Previous Work in Hierarchical Clustering with ART Networks

We have discussed in the previous section how variations of the vigilance parameter result in various structural views of the data. Since these views correspond to different levels of abstraction of the data, vigilance variations can be exploited to detect hierarchical structure in data. Researchers in areas other than text clustering have previously explored this approach. For example, Ishihara *et al.* [19] proposed "arboART", a series of ART networks processing data at different vigilance and in which each network receives as input the output of the preceding level. Some of the work by Bartfai and White [20] follow the same approach, as we will see briefly. Lavoie *et al.* [21] follow a different avenue using Fuzzy ART with a new selection function for the winning neuron. Since our current work is based on ART1, this approach to form hierarchies will not work, and we will therefore limit ourselves to variations of vigilance and linked networks. This being established, there are essentially two possibilities to form hierarchies with ART1 networks. The first consists in choosing the output neurons of multiple networks working at different vigilance on the same dataset [22]. We recall that the output neurons correspond to the clusters; therefore output neurons obtained at a given vigilance can be used as the nodes of a topics tree. A network working at minimal vigilance will produce the roots of the hierarchy (indeed, it is a forest and not a single tree that is formed) while categories obtained at successively higher vigilances will form the subsequent levels. Fig. 2 illustrates a simple case with two ART networks.

Two nodes $C_{k,i}$ and $C_{k+1,j}$ on two adjacent levels k et k+1 are linked if $C_{k,i} \cap C_{k+1,j} \neq \{\}$. The intersection here means to verify if there are any common documents in the categories $C_{k,i}$ and $C_{k+1,j}$ represented by the corresponding nodes. In other words, a hierarchy going from general (top) to specific (bottom) can be constructed by linking clusters that have documents in common on two subsequent levels.

The second approach to built hierarchies with ART is the serial interconnection of networks [19], [23]. The first network of the chain processes data at high vigilance, which allows the creation of many specific clusters, thus forming the lowest level of the hierarchy. The prototypes of this network are then used as inputs of the next network. Hence, the networks in the chain successively construct more general levels of the hierarchy by fusing the previous level clusters. As such, this approach is an instance of traditional agglomerative hierarchical clustering algorithms. A simple example of this architecture with two ART networks is shown on Fig. 3. We put the emphasis on the fact that with this approach, networks subsequent to the first one do not process the actual document data but the prototypes representing group of similar documents discovered by preceding networks. Vigilance value must be determined for each network of the series and is not necessarily in decreasing order. Clusters formed by the j^{th} network of the chain are used to form the $(h\text{-}j)^{th}$ level of the hierarchy (h is the height of the hierarchy, with level 0 being the root).

There is actually a third possible approach to hierarchical clustering with ART networks. Again, a series of ART network is used but this time a divisive algorithm is implemented [23]. In this case, general clusters are first formed and are then successively divided into smaller specific groups. The difference between an input and the prototype to which this input was assigned is propagated to the next level. An additional parameter called resolution ensures that the next module is activated only if the difference is large enough. Four sets of connections exist between each module of the chain. Since this architecture is more complex than the other two described previously and since Bartfai and White have shown that it gives similar clustering quality, we will not test this architecture here.

The application of these ideas to hierarchical text clustering has been limited to Vlajic and Card [10] and our own preliminary exploration work a few years ago [11]. In the latter case, the hierarchy is assembled manually based on clustering results obtained at various vigilance levels on a very small text collection. The quality evaluation is subjective. Vlajic and Card used a

non-binary version of ART (known as ART2) modified to handle web pages. Here again a very small set of 20 web pages is used and no quantitative quality evaluation performed. This is the type of situation we want to avoid in this work by clustering a relatively large, benchmark text collection and evaluating quality quantitatively with a proven quality measure.

Experimental Settings

To evaluate the abilities of ART networks to discover hierarchical structure in text, we employ the following experimental strategy: First, we form a hierarchy with each of the two approaches described in the previous section using the exact same text data to ensure comparability; second, we measure the clustering quality objectively at each level and observe structural qualities (such as number of sub-clusters at each level and number of undivided clusters between levels). We then compare the quality of each level of the ART-produced hierarchy with the quality obtained using k-means and a conventional hierarchical agglomerative clustering (HAC) algorithm implementing the minimum variances criterion (Ward's method). Again, the exact same text data was utilized to ensure comparability of results. Although k-means is a partitioning algorithm giving flat clusters, the clusters for various values of k can be used to form a hierarchy similarly to what we have done with independent ART networks. Lastly, we evaluate clustering visually to determine how useful hierarchies built with ART would be to an actual human user.

The text data used is the Reuter 21578 benchmark corpus, which is transformed into the standard bag-of-words vector space representation. Well-established pre-processing is applied, such as stop words removal and dimensionality reduction by removing words that do not exceed a minimal occurrence frequency. We removed as many words as possible before zeros-only document vectors started appearing, which corresponds to removing words appearing in 77 documents or less. This had resulted in the best quality in our previous experiments and also the fastest processing with N=357 features. Objective quality evaluation is conducted by computing the F_1 external validity of the clustering solution at each level. This means we compare the clustering solution $C = \{C_i \mid i = 1, 2, \ldots, M\}$ to a *given* or *desired solution* $S = \{S_j \mid j = 1, 2, \ldots, M^s\}$, hence measuring the ability of the clustering algorithm to retrieve the given desired solution prepared by human classifiers, which is assumed to be the ground truth. The clustering solution C is a set of *clusters* C_i while the desired solution S is a set of *topics* S_j. Both C_i and S_j are sub-sets of

D = {d_0, d_1, … d_R}, the set of documents to cluster. This manner of computing quality has been used successfully in clustering before (see for instance [24], [25]). Based on [24] and [25], F_1 is given by:

$$F_1 = \frac{\sum_{j=1}^{M^s} |S_j| F_{1j}^*}{\sum_{j=1}^{M^s} |S_j|} \tag{6}$$

Better quality is achieved with higher F_1 values, in the range [0,1]. F^*_{1j} is the F_1 value of the cluster that best matches topic j in terms of maximizing its F_1 value. The F_1 value of a cluster i with respect to a fixed topic j is:

$$F_{1i} = \frac{2\alpha_i}{2\alpha_i + \beta_i + \chi_i} \tag{7}$$

where $\alpha_i, \beta_i, \chi_i$ are given by equation 8:

$$\alpha_i = |C_j \cap S_i| \quad \text{i.e. the number of true positives} \tag{8a}$$

$$\beta_i = |C_j| - \alpha_i \quad \text{i.e. the number of false positives} \tag{8b}$$

$$\chi_i = |S_i| - \alpha_i \quad \text{i.e. the number of false negatives} \tag{8c}$$

We note that equation 7 is obtained by simple algebraic manipulations from the well known F_1 effectiveness measure of information retrieval and text classification [26], [27]:

$$F_b = (b^2+1)pc / [b^2p+c] \tag{9}$$

where :

$$p = \alpha / (\alpha + \beta) \text{ is the } \textit{precision} \text{ ; and} \tag{10}$$

$$c = \alpha / (\alpha + \chi) \text{ is the } \textit{recall}. \tag{11}$$

Parameter *b* determines the balance between precision and recall and its value is usually set to 1, which is what we have done to derive equation 7. In text classification, the number of true

positives, false positives and false negatives are not computed exactly as in clustering (Eq. 8) since one has *a priori* knowledge of which class corresponds to which topic in the ground truth solution. Details of the differences between the text classification and text clustering F_1 computation are presented in [25].

Some authors elect to compute a global quality for the whole hierarchy [5], [28]. Such an approach to quality computation allows one to take into account the *degree of error*. For example, misclassifying a document about camping under sports rather than outdoor activities is less dramatic than classifying it under Computer Science. We consider in our case that this does not serve our purpose of evaluation since it may unfairly inflate quality: we prefer a stricter definition of quality to establish an un-inflated baseline quality. Noting and averaging quality on each of the few levels gives a good idea of the hierarchy quality, which is sufficient to achieve our goal and this is therefore what we will do.

Some may object to the choice of F_1 as a quality metric. F_1 is one measure of clustering quality among many others, it has its strengths and limitations but one must keep in mind that there is no single perfect way to measure clustering quality. We are not the first to use F_1 to establish clustering quality, as mentioned previously. As such we merely use one existing measure of quality among the large set available. Our goal is not to re-invent the wheel and propose yet another clustering evaluation methodology, but rather we prefer to use an existing one for compatibility and comparability with our previous work. Our choice of external validation Vs internal is justified by the need to establish usability of the resulting hierarchy by humans. We therefore made a choice of measure that involves the presence of a "ground truth" solution. We concur that this is not the only solution as users may have different views of how documents shall be grouped. However, this comparative evaluation with a given solution is a simple, low cost way to evaluate quality and it serves our purpose. Furthermore, the literature in both text clustering and supervised text categorization is rich in examples of evaluation using ground-truth comparative approaches. Indeed, doing so one actually measures how well the algorithm re-discovers one specific human crafted solution. This is the reason why we also built a user interface so that clusters can be viewed and evaluated subjectively by humans (and this is actually how we identified the most problems with the clusters).

For the independent ART1 networks approach, we used clustering outputs of previous experiments with the Reuter collection. The vigilance values used are 0.04, 0.06 and 0.1 for the intermediary and inferior levels of the hierarchy. For roots, clustering at minimal vigilance (0.005) was used. For ART networks in series, we used the exact same dataset (same documents, same pre-processing) as for independent networks to ensure results are comparable. In this case however, vigilance values are not known in advance: we must experimentally find the ones that work best.

Experimental Results and Discussion

Results with independent ART1 networks

We have formed a four-level hierarchy by linking the clusters formed by four ART networks functioning independently with different vigilance values. Vigilance values for each level were selected among the best qualities obtained during previous experiments with flat clustering, thus there is no surprise in terms of quality as we choose the highest quality solutions as building blocks for the hierarchy. However, within a real application, a fundamental problem would emerge: quality being unknown at first, the *a priori* choice of vigilance becomes problematic, as we have pointed out in other work [13]. For k-means and the HAC algorithm implementing the minimum variances criterion, we make two observations. First, we measure the quality obtained for the levels with the same number of clusters as the ones used for ART. Secondly, we also consider the average quality over the whole range (from 45 to 200 clusters) as it may be unfair to k-means and HAC which must compete with the best pre-determined levels used for ART.

Tables 1 and 2 summarize the main characteristics of the hierarchy obtained with independent ART1 networks. The hierarchy has 45 roots, corresponding to the minimal number of clusters achievable at minimal vigilance. Table 2 shows the average number of sub-clusters, that is the number of children clusters attached "under" a cluster at a given level. Fig. 4 shows the quality obtained with k-means and the HAC algorithm implementing the minimum variances criterion. With ART, the average quality over the levels of the hierarchy is $F_1 = 0.37$, while it is 0.22 and 0.24 for k-means and for HAC respectively. Of note is that with the same dataset as used with ART1, both k-means and the HAC algorithm give (in the range of 45 to 200 clusters) a quality F_1

≤ 0.27, which ART1 exceeds globally and for each level of the hierarchy. The quality decreasing with the number of clusters, increasing the number of clusters further for k-means and HAC would only make their case worse. It is therefore not possible to get a better quality hierarchy with k-means or with HAC than what was achieved with ART (keeping in mind the limitations of our quality metric, and for that matter of any quality metric).

Level	**Vigilance**	**F_1**	**Nbr of clusters**
0	0.005	0.40	45
1	0.04	0.38	64
2	0.06	0.31	94
3	0.1	0.38	231

Level	**Avrg # of sub-clusters**	**Std dev**	**# of undivided sub-clusters**
0 to 1	3.5	4.0	19 of 45 (42%)
1 to 2	5.2	7.0	27 of 64 (42%)
2 to 3	7.6	9.8	27 of 94 (29%)

One advantage of this architecture is the parallel processing of data for each level since different ART networks can each produce their output independently. Then a post-processing module can assemble them into a hierarchy. Two problems were noted, however.

The first problem is that a large number of clusters did not split when going from one level to the next. Between levels 0 and 1 and then levels 1 and 2, 42% of the clusters had only a single sub-cluster. Between the final two levels, 29% of the clusters did not split (see Table 2). In these cases, the unique sub-cluster is identical to the parent cluster, which is contrary to the objective of the hierarchy, namely to divide the informational space to facilitate the search of information. This aspect gives another point of view on the hierarchy's quality. We believe that in order to make the hierarchy truly useful, this problem should be solved to limit single child clusters to exceptions.

On the other hand, only a few clusters are separated into a high number (≈ 40) of sub-clusters, but forty-odd sub-clusters appear to be reasonable. This evaluation of the number of sub-clusters is arbitrary. Determining the number of sub-clusters that are acceptable to a user is a

matter of research on human-machine interfaces, a topic not addressed in this paper. The same goes for hierarchy depth. Scientific literature we have reviewed on both supervised and unsupervised learning of document hierarchies is generally silent on this topic. We could, however, raise the following references for Reuter, but they do not in any way address the ergonomic aspect: with a previous version of Reuter (Reuter-22173), Koller and Sahami [4] built a two-level hierarchy with three and six clusters per level, and Weigend *et al.* [29] with five clusters on the top level. With the same Reuter collection as we used, D'Alessio et al [30] instead used the five meta-categories that came with the collection as top-level topics in the hierarchy. In these cases, there are far fewer categories per level than what we found with ART.

In addition, we observed what is done with the topics hierarchies built by human classifiers for the Internet, such as Yahoo (http://dir.yahoo.com/) and Google (http://directory.google.com/). In such cases, there is a certain structural similarity between the ART hierarchies, such as, for example, the depth (approximately five levels) and the number of subclasses: between 10 and 50 at the higher levels. In some exceptional cases, there are more subclasses but usually fewer and fewer as the topics become more specialized. This leads us to say that although ART generates far too many undivided clusters, the hierarchy extracted from the data seems to have reasonable structural properties of depth and number of subclasses, similar to the ones one can find in commercial-grade, human-built hierarchies.

The second problem with the independent ART network architecture is the lack of consistency, i.e. that a k+1 level sub-cluster can be the child of more than one k-level cluster. Consequently, the sub-clusters do not form documents subsets of their parent cluster. In the hierarchy built by ART, a sub-cluster has on average three different parents (standard deviation of 2.8). In some cases, there are even up to eight and eleven different parents. The cause of this phenomenon is precisely the independent operation of the networks: the clusters formed at various vigilance values are created with different criteria, thus providing a totally different viewpoint of the data structure. Then, the clusters at two levels are linked merely on the presence of at least one common document. Clusters obtained through low vigilance would thus not necessarily correspond to partitions of clusters formed at higher vigilance. Accordingly, they would not be subsets of clusters formed at high vigilance. A possible solution to this problem would consist of adding some dependence between levels by not rolling back the weight of connections to zero

between each level of vigilance (if using a common network) or by communicating the weights obtained by the processing of one network to another network.

Before moving further into this area, which considers the absence of so-called consistency as being problematic and to come up with solutions, one must first question the gravity of the problem. The idea of a document having more than one parent is apparently undesirable, as users could easily get lost in such a hierarchy. For example, if the user wants to move up in the hierarchy following an unsuccessful search, the user would have more than one option, whereas when she was at first moving down the hierarchy there was a single option. In reality, this is not necessarily a problem with, on average, only three parents as we have here. The user could move up the hierarchy using the same path by remembering the descending path using a stack (such as the "history" button in Web browsers).

Furthermore, having the ability to access a document from more than one parent may be very beneficial, as it would provide different ways of accessing the same information. From a semantics standpoint, this situation is reflected as a subtopic that "belong" to several different topics. For example, there could be a topic "gold," which might be a subtopic of "precious metals" and of "electronics." This could result from the fact that gold is a precious metal and is used in electronics components. Therefore, the hierarchy does in fact correctly represent, from a structural standpoint, the relationship between the subtopic and the parent subjects. However, each link has a different meaning. In one case, the subtopic is linked to the topic through a "type of" relationship (gold is a *type* of precious metal) while, in the other, it is through a "use" relationship (gold is used in electronics). We also note that it is exactly what the Yahoo and Google hierarchies do and it is one of the objectives of soft clustering. In short, consistency appears to not be a problem after all if the number of parents is not too high.

Results with ART1 networks in series

We know from the previous experiments with independent networks that $\rho = 0.1$ gives M=231 cluster, which seems a reasonable number of leaves for a hierarchy (we want to avoid an overload of information for users by providing too many clusters to explore). Since ART networks in series are a bottom-up approach to hierarchical clustering, we will start with this maximum number of clusters at the bottom of the hierarchy. For the upper levels, again, we were faced with the problem of finding the appropriate vigilance and thus preceded by trial and error to find acceptable values of vigilance. We have found that $\rho_{min} \leq \rho \leq 0.5$ was giving good results (with minimal vigilance $\rho_{min} = 0.005$). In our experiments, we used both extremes of this vigilance interval, but any other values within the extremes should be satisfactory. In practice, the vigilance values selected will be determined by the application, depending on the required number of clusters at each level.

It was not possible to form more than three levels in the hierarchy. Indeed, whatever the vigilance value at the top level, the clustering obtained was always identical to the one on the second level, which indicates that no further clustering was possible. Table 3 summarizes the characteristics of each level of the hierarchy. When using the ART serial network approach, there was no inconsistency such as observed with the independent networks as the clusters themselves were successively amalgamated at each level. The average quality for the entire hierarchy is $F_1 = 0.34$, slightly lower than with independent networks. Only at vigilance 0.005 is the quality lower than the maximum values obtained with k-means and HAC, but on average this type of ART-produced hierarchy also offers higher F_1 quality than k-means and HAC.

The column "Stand. dev." in table 3 refers to the standard deviation in the number of sub-clusters. For $\rho = 0.005$, this value indicates a wide variation in the number of sub-clusters, a variation larger than what we observed for independent networks. Indeed, at that vigilance value, the four clusters with the most sub-clusters have respectively 121, 44, 11 and 3 sub-clusters. Then, three clusters have only two sub-clusters and the 39 other clusters (39/46=85%) have only one (i.e. undivided sub-clusters).

ρ	Level	# of clusters	F_1	Avrg # of sub-clusters	Stand. dev.
0.005	0	46	0.25	22.5	13.4

0.5	1	101	0.39	2.2	3.4
0.1	2	231	0.38	n/a	n/a

Hence, like the independent networks, the serial networks result in many clusters that are not divided from one level to another. In fact, there are many more: 85% and 74% between the upper tiers. That is where there is a major problem since the hierarchy does not excel in its role in partitioning document space. This situation is the result of too few active components (those with a value of 1) in the prototypes, which are the intersection of all document vectors assigned to the cluster a prototype represents. Since few components are active in the prototypes, when they become the inputs of the next network, there is only a very low probability of intersection between the inputs. It thus becomes very difficult to find clusters among the data with few or no common attributes. Accordingly, the clusters have little chance of splitting further into sub-clusters.

The small number of active components also limits the height (the maximum number of levels) of the hierarchy. In the course of our experiments, it was impossible to form more than three levels. The maximum number of levels h that could be formed with the serial architecture was studied by Bartfai [23] and is determined by the magnitude of inputs $K = ||\mathbf{x}||$ and the vigilance ρ:

$$h = \left\lfloor -\frac{\log K}{\log \rho} \right\rfloor + 1 \tag{12}$$

Bartfai presumes in the calculation that K and vigilance ρ are constants, which they are not in our case and which is not realistic. Let us look at and delve deeper into Bartfai's reasoning. The fundamental principle that should guide the approach to determining the maximum height of the hierarchy is the progressive erosion of prototypes at each successive level. In fact, the magnitude of the prototypes $||\mathbf{t}||$ monotonically decreases with time in a ART1 network following updates by intersection:

$$\mathbf{t}' = \mathbf{t} \wedge \mathbf{x} \tag{13}$$

Thus, the new prototype $\mathbf{t}'$ is the intersection of the current prototype $\mathbf{t}$ and the input $\mathbf{x}$ with which it has passed the vigilance test. This means that:

$$|| \mathbf{t}' || \leq || \mathbf{t} || \tag{14}$$

For example, if $\mathbf{t}$ = [1 0 1 1] and $\mathbf{x}$ = [1 1 0 0], then $||\mathbf{t}|| = 3$, $||\mathbf{x}|| = 2$ and $||\mathbf{t}'|| = ||\mathbf{t} \wedge \mathbf{x}|| = 1$, which will necessarily be smaller or equal to $||\mathbf{t}||$.

In the context of a serial ART network, a prototype $\mathbf{t'}^n$ of level *n* would be the input $\mathbf{x}^{n+1}$ of the next level. The magnitude of this input will again be reduced by the processing of the ART network at level n+1. Therefore,

$$|| \mathbf{t}'^{n+1} || \leq || \mathbf{t}'^{n} || \tag{15}$$

It thus becomes essential to have prototypes that are of a sufficient magnitude as entries at level n+1, otherwise, it would be impossible to further split them into a smaller number of clusters. This is the cause of the maximum number of levels possible. The absolute lower bound of a prototype's magnitude can be established as $||\mathbf{t}'^{n+1}||_{min}$ below which no new level can be formed. In fact, when the largest prototype will be of magnitude 1, it becomes impossible to split the prototypes even further. Note that, in practice, the bound will be reached beforehand, i.e. for a magnitude of $||\mathbf{t}'^{n+1}||'_{min} \geq 1$. The reason for this is that formation of new clusters is dependent upon all inputs and their potential for intersecting each other, as a prototype $\mathbf{t}_j$ is the global intersection of all inputs $\mathbf{x}_i$ that are assigned to cluster C_j ($\mathbf{t}_j = \bigcap_{x_i \in C_j} \mathbf{x}_i$). However, the absolute lower bound below which no new cluster can form in theory remains:

$$|| \mathbf{t}'^{n+1} ||_{min} = 1 \tag{16}$$

Now, we must calculate the magnitude of the inputs at each level. We re-write the vigilance test by moving the denominator from the left to the right side of the inequality for an input $\mathbf{x}^n$ and a prototype $\mathbf{t}^n$ at level *n*:

$$|| \mathbf{t}^n \wedge \mathbf{x}^n || \geq \rho || \mathbf{x}^n || \tag{17}$$

From Eqs. 13 and 17, we obtain the updated magnitude of the prototype vector $\mathbf{t}^{n}$' following processing by the ART network at level *n*, which will be:

$$|| \mathbf{t}^{n}' || \geq \rho || \mathbf{x}^n || \tag{18}$$

and since $\mathbf{t}^{n}$' will be the input $\mathbf{x}^{n+1}$ of the next level, we could also write:

$$|| \mathbf{x}^{n+1} || \geq \rho || \mathbf{x}^n || \tag{19}$$

We recursively apply Eq. 18 for each level:

$$|| \mathbf{t}^h || \geq || \mathbf{x}^1 || \prod_{n=1}^{h} \rho^n \quad (20)$$

$||\mathbf{x}^1||$ must then be the magnitude $||\mathbf{x}^1||_{max}$ of the largest input to ART 1 network since it offers the most possibilities of partition into sub-clusters. We thus obtain a lower bound for $||\mathbf{t}^h||$ with the best possible initial conditions.

Lastly, we apply Eq. 16 as termination condition to find *h* with equation 20:

$$h = \min(n: || \mathbf{x}^1 ||_{max} \prod_{n=1} \rho^n = 1) \quad (21)$$

We are thus looking for the minimal *n* that ensures that $|| \mathbf{t}'^n ||_{min} = 1$ (Eq. 16).

In our experimental situation, among the 3,299 documents Reuter submitted to ART 1 network, the greatest of $||\mathbf{x}^1||_{max}$ is 95. We thus obtain the following expansion of equation 21:

n =1: $95\ \rho^1 = (95)(0.1) = 9.5 > 1$, continue;

n =2: $95\ \rho^1\ \rho^2 = (95)(0.1)(0.5) = 4.75 > 1$, continue;

n =3: $95\ \rho^1\ \rho^2\ \rho^3 = (95)(0.1)(0.5)(0.005) = 0.024 < 1$, stop.

$\therefore\ h = 3$

We note that if $\rho^3 > 1/4.75$, we would obtain $h = 4$. This is caused by the inaccuracy of the bound set by Eq. 16 used as termination condition in Eq. 20. In fact, remember that, in practice, the bound will be reached for a magnitude $||\mathbf{t}'^n||'_{min} \geq 1$ since formation of new clusters is dependent on all inputs and their potential for intersection among themselves. Thus, we need to determine $||\mathbf{x}^n||_{max}$ at each level, not only at the first level. However, even in this case, it is not possible to predict all interactions between the inputs. For example, in our case, we have the following $||\mathbf{x}^n||_{max}$:

n = 1: $||\mathbf{x}^1||_{max} = 95$

n = 2: $||\mathbf{x}^2||_{max} = 13$

n = 3: $||\mathbf{x}^3||_{max} = 13$

n = 4: $||\mathbf{x}^4||_{max} = 3$

At the output of the level 3 network, thus at the input to level 4, $||\mathbf{x}^4||_{max} = 3$. In theory, a vigilance level > 1/3 should enable the formation of the fourth level. Our experiments have revealed that such is not the case. The explanation is that no prototypes formed at level 3 have common components, which prevents any additional partitioning regardless of the level of vigilance. The maximum number of levels in a hierarchy is ultimately determined by the

progressive erosion of prototypes between the levels, thereby necessarily leading to low potential for intersection and thus a very low probability that an additional level composed of different clusters will be formed.

Hierarchy visualization

We wanted to verify whether the hierarchies created by ART were actually useful in navigating document clusters. This was achieved by rendering the hierarchy graphically and exploring the documents collection with the hierarchy. Although the F_1 quality is not extraordinarily high; we have shown in previous work that on average it exceeds the clustering quality obtained with k-means. This being said, it would be interesting to see if the clusters are in fact useable for finding information. To do this, we added a post-processing module to the ART networks, which builds a hypertext (HTML) representation of the hierarchy. K-means and HAC were not evaluated visually because we were only interested in observing if ART1, given its superior F_1 quality would actually result in a hierarchy of documents that is usable. Furthermore, this exercise was completed solely for independent networks that, from the outset, result in fewer undivided clusters than serial networks and a slightly higher F_1 quality. It has also allowed us to experiment with the consistency "problem" (multiple parents). Fig. 5 shows the graphical user interface used to display and explore the hierarchy. The evaluation conducted here was anecdotal and incomplete since we have considered only a very small part of the hierarchy. Nevertheless, this exercise allowed us to identify important usability and quality issues. A more comprehensive evaluation with multiple users is a possibility for future work but was not realizable within the scope of this work.

The first observation following the use of this graphical interface is that certain clusters are characterized by a single, very vague term that conjures up no topic information that could be used by a user. For example, the first five clusters respectively have "year," "reuter," "said," "vs," and "says" as unique attributes of their prototype. Most of these are words used very frequently in the collection. Table 4 shows the most frequently used words in all of the data. Four of the so called "problem words" in the example are present (shaded). The word "says" does not appear in the table as it has a much lower frequency, appearing 193 times in the collection and present in 186 documents.

Word	Total number of times word appears in the collection	Number of documents containing this word
said	7279	2026
mln	5928	1610
vs	5052	966
dlrs	3900	1451
reuter	3019	3005
pct	2759	974
lt	2545	1864
cts	2523	960
net	2451	988
billion	1935	597
year	1771	854

Some of the clusters formed with these words are highly populated and intersect with a large number of desired topics, thereby greatly affecting the overall quality. Such is the case with the first two clusters (those with "year" and "reuter" as unique attribute), each respectively containing 781 and 1,502 documents distributed among 74 of the 93 topics of the human crafted solution. From this standpoint, these two clusters are truly distinct and constitute catchalls compared to other clusters. Fig. 6 provides a good illustration of this situation by showing part of the cross-correlation matrix between clusters (rows) and topics assigned by human classifiers (columns). The number in each cell indicates the number of common documents i.e. those correctly classified. Only these two clusters have non-zero values (and also quite high values) in almost every cell.

Moreover, in the cross-correlation matrix in Fig. 6, cluster 0, whose unique attribute is the word "year," is dominated by 91 documents from topic 16 ("acquisitions") and 109 documents from topic 25 ("earnings"). Nevertheless, it still only amounts to 91 out of 719 documents and 109 of 1,087 documents actually assigned to each of these topics in the desired solution. Cluster 1, however, whose unique attribute is the word "reuter," is more strongly dominated by 480 documents from topic 16 ("acquisitions") out of a total of 719 documents (this is 67%, making cluster 1 a fairly large container of true positives for the topic).

The strong presence topic 16 in cluster 1 may not seem surprising at first, since it is the second most populated Reuter topic. A document therefore has a relatively high probability from the outset of finding itself in this topic, and a cluster should have a relatively high probability of

containing a fair number of documents of this topic. A given document selected randomly from the 3,299 documents of the data set has a probability P(ACQ) = 719/3299 = 0.22 of being labelled with topic "acquisitions" in the desired solution. Then, cluster 1, which is characterized by the unique attribute "reuter," contains 67% (480/719) of documents from topic "acquisitions". This seems a high proportion of documents. One may wonder if cluster 1 is just a statistical accident, given the unlikely and unrepresentative reuter feature. What is the probability that a document chosen at random is in class "acquisition" and is placed in cluster 1? Since there are 45 clusters at level 1 of the hierarchy, there is a P(G1) = 1/45 = 0.02 chance of putting a document into cluster 1. For the first run, the conjunctive probability of the two events will be P(ACQ) * P(G1) = 0.22 * 0.02 = 0.0045. The second question that arises is: what is the probability that as many documents (480) from class "acquisitions" can be randomly assigned to the same cluster 1? This has a probability of $< 10^{-30}$. It is thus very unlikely that the clustering obtained for cluster 1 is a statistical artifact. ART1 is doing something sensibly non-random. Still, what is very peculiar is that the word "reuter" is used as a critical characteristic for this cluster. This word appears to simply be a source identifier and therefore should not be semantically determinant for topic "acquisitions". A possible explanation is that while other words may have had an influence on the attribution of documents to this cluster during processing, the word "reuter" being very prevalent in the collection, it was the only word common to all documents in the cluster and hence the only word left following the multiple updates to the prototypes.

Cluster #	**Prototype keywords**	**Number of documents for topic "acquisitions"**
7	acquire	7
31	merger	2
33	purchase assets	14
34	recent takeover	9
39	acquisition	1
42	bid	45

The second observation we made following the use of the graphical interface is that 90% of the documents of topic "acquisitions" are distributed among seven clusters, namely clusters 1, 7, 31, 33, 34, 39 and 42. It thus appears that these clusters may fragment this topic, forming "acquisitions" subtopics, which could be quite interesting. By viewing each, we instead find that several of these supposedly subtopics of "acquisitions" contain very few documents that simply

contain a word that is synonymous to or semantically related to “acquisitions”. Table 5 shows the number of documents and the prototype keywords of these clusters. Because they do not match lexically with one another, these keywords have forced the formation of separate clusters for topic “acquisitions.”

Words such as “year,” “reuter,” “said,” “vs,” and “says” (among others) that appear to be the root of the first problem are not eliminated during pre-processing with filtering by minimal frequency, but they may be eliminated when filtering with TFIDF. Furthermore, terminological standardization could help solve the topic fragmentation problem at least for words of the same family (e.g. “acquisition” and “acquire”). In previous work, we saw that these two techniques do not necessarily help increase F_1 quality but they have the potential to improve the user’s experience by solving some of the problems we just observed. We have thus created a hierarchy with a text dataset on which terminological standardization was applied (essentially a suffix stripper) and feature selection completed with TFIDF rather than the simple minimum frequency of occurrence threshold used previously.

Unfortunately, our experiments with TFIDF and terminological standardization have solved none of the problems. The hierarchy formed with TFIDF and terminological standardization still experience the fragmentation problem with class “acquisitions.” Now, we have the stems “acquir” and “acquisi,” which do not allow “acquisitions” to be standardized with “acquire.” Semantically related words, such as “merger” and “takeover” still form separate clusters. Keyword “Reuter” was removed from the vocabulary by the TFIDF selection, but several other problem words, such as “year” and “said,” were not and continue to cause problems.

Based on the observations from the previous paragraph, we explored another potential solution to eliminate the words that cause a problem: we have treated problem words as *domain stop words*. We inserted these words in the list of general stop words (with prepositions, articles, etc.) to force their exclusion as attributes during pre-processing. This approach constitutes a significant human intervention, which partially defeats the initial purpose of clustering, which is exactly to proceed without human intervention. However, in a real-life application, it is possible that it would be worthwhile to include this additional step if the usability and overall quality of the hierarchy becomes greatly improved. The words removed were:

said	year	pct	vs	dlr
say	today	lt	mln	usda

do	cts	sees	see	dlrs
does	billion	reuter		

The criterion for word exclusion was based on the author's impression of the utility of these words as representative features for cluster prototypes. The result of this operation is again unsuccessful. First, F_1 sustains a decrease of about 10%, which leads one to think that some of the words we removed indeed have a role in the overall quality of clustering. For example, we removed words such as "dollars" (in its many forms), which could in fact be globally useful even if our first impression was to believe that the clusters formed with these words were invalid. Secondly, it appears that other words have taken the place of the problem words that were removed, resulting in new problems (the word "ar" for example, is used as unrepresentative prototype feature). It may require several attempts at manipulating the vocabulary to arrive at something that is acceptable, but we have not judged worthwhile to explore this avenue since it seemed unrealistic to perfect such heavy-duty vocabulary manipulation when one of the objectives of unsupervised techniques like clustering is to avoid human intervention.

With respect to the problem of the superfluous formation of sub-clusters because of words that are semantically related, it is one that can be expected given the nature of natural language text that may not always convey the same ideas with the same words across documents assigned to an identical topic. Rather, synonymous words or other semantically related words to express the same ideas are employed. Consequently, clustering will not necessarily form the clusters one would expect and clusters will not necessarily be at the same level of abstraction as a solution predetermined by humans. Thus, ART can (like any other clustering algorithm) discover clusters that generalize or specialize desired topics. A generalization is a cluster that includes two or more topics while a specialization is when two or more clusters split the documents of a single topic (as we have seen in the case of topic 16 ("acquisitions"). Rarely would one obtain a perfect generalization or specialization, i.e. without the presence of documents originating from other topics. As illustrated in the cross-correlation matrix in Fig. 6, we could thus expect a mixture of topics that has a negative impact on a user's experience and on quality.

Conclusions and Future Work

The main contribution of this work is to have investigated in details the unsupervised learning of topics hierarchies with ART1 neural networks, a poorly studied hierarchical text clustering algorithm. Our experimental methodology based on the proven F_1 quality measure, benchmark Reuter 21578 corpus, standard bag-of-words vector space representation and well-established pre-processing allows for easy reproduction of our work and comparison with other text clustering results [25].

We have put two different approaches to the test: independent ART networks and serial ART networks. We identified the strengths and weaknesses of each approach. In both cases, we encountered the problem of finding the appropriate vigilance value for each level of the hierarchy. Another problem common to the two approaches, albeit more important for serial networks, is the significant number of clusters that did not split when moving down from one level to the next. Many clusters with one unique child are problematic for hierarchies as they are contrary to their objective, which is to divide the information space to facilitate the search of information. This aspect provides another insight into the quality of the hierarchy. We believe that this is an important and very useful discovery that, unfortunately, highlights a shortcoming of the techniques investigated in this paper.

As well, we have established that the average F_1 quality of hierarchies built with both independent ART networks and serial ART networks exceeds the F_1 quality of k-means and of a conventional hierarchical agglomerative clustering algorithm implementing the minimum variances criterion. We are currently performing a comparative study of ART with a UPGMA cluster similarity HAC which Steinbach *et al*. showed performed best among HAC algorithms [3]. Our objective is to determine whether this algorithm exhibits the same problems as the hierarchies build with ART and whether it offers a higher F_1 quality.

Furthermore, we have confirmed Bartfai's results to the effect that the number of levels is limited with serial networks, while they can be defined arbitrarily for independent networks. We have reviewed and improved on Bartfai's explanation for the calculation of the number of possible levels in the specific context of text data.

In addition to objective, quantitative evaluation with F_1, we performed a series of qualitative evaluations of the ART-produced hierarchies through visualization. For independent networks, we established that the problem of multiple parents (inconsistency) is not really a

problem as long as the number of parents is restricted. In fact, it is an advantage that allows access to the information in different ways. We found that the topic hierarchies built by human classifiers for the Internet, such as Yahoo and Google, offer a certain structural similarity to the hierarchies built with ART. This concerns multiple parents in particular, obtained with independent networks, but also with respect to the restricted number of levels and the number of children. Moreover, the visual evaluation allowed us to see that the F_1 quality does not always tell us everything about the quality of a user's information search experience. For instance it identified two additional problems with ART-built hierarchies: first, certain words are inadequately chosen as attributes of the prototypes; and second, certain topics are split by ART because of semantically related words. We found that automatic pre-processing with TFIDF and terminological standardization does not solve these types of problem. We also tested forced elimination of what we deemed domain stop words as a final solution to this predicament without success. The use of dictionaries or linguistic analysis techniques could help solve this difficulty by allowing for the identification of the relationships between the words (Hotho *et al.*, 2003). We plan to investigate such semantic feature selection and clustering approaches in future work.

References

[1] Koller D, Sahami M (1997) Hierarchically classifying documents using very few words, Proc. of the 14th International Conference on Machine Learning (ICML97), pp 170-178.

[2] Kiritchenko S, Matwin S, Nock R, Famili F (2006) Learning and Evaluation in the Presence of Class Hierarchies: Application to Text Categorization, Proc. Of the Canadian Artificial Intelligence Conference, Québec, Canada.

[3] Steinbach M, Karypis G, Kumar V (2000) A Comparison of Document Clustering Techniques, Proc. Of the Sixth ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Boston, MA, USA.

[4] Heuser U, Rosenstiel W (2000) Automatic Construction of Local Internet Directories using Hierarchical Radius-based Competitive Learning, Proc. of the 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2000) July 23-26, 2000, Orlando, Florida, Volume IV (Communications Systems and Networks), pp. 436-441, invited paper.

[5] Zhao Y, Karypis G (2005) Hierarchical Clustering Algorithms for Document Datasets. Data Mining and Knowledge Discovery, Vol. 10, No. 2, pp. 141-168.

[6] Fung BCM, Wang K, Ester M (2003) Hierarchical Document Clustering Using Frequent Itemsets. Proceedings of the SIAM International Conference on Data Mining, SDM'03, San Francisco, CA, pp 59-70.

[7] Kummamuru K, Lotlikar R, Roy S, Singal K, Krishnapuram R (2004). A hierarchical monothetic document clustering algorithm for summarization and browsing search results. Proceedings of the 13th international conference on World Wide Web, pp. 658-665.

[8] Freeman RT, Yin H (2004) Adaptive topological tree structure for document organisation and visualisation, Neural Networks, v.17 n.8-9, p.1255-1271.

[9] Grossberg S (1976) Adaptive pattern classification and universal recording : I. Parallel development and coding of neural feature detectors, Biological Cybernetics, Vol 23, pp 121-134.

[10] Vlajic N, Card HC (1998) Categorizing Web Pages using modified ART. Proceedings of IEEE 1998 Canadian Conference on Electrical and Computer Engineering, Waterloo, Canada.

[11] Massey L (2002) Structure Discovery in Text Collections, Proceedings of KES'2002, Sixth International Conference on Knowledge-Based Intelligent Information & Engineering Systems, Podere d'Ombriano, Italy.

[12] Massey L (2003) On the quality of ART1 text clustering, Neural Networks(16)5-6, pp.771-778.

[13] Massey L (2005) Real-World Text Clustering with Adaptive Resonance Theory Neural Networks, Proceedings of 2005 International Joint Conference on Neural Networks, Montréal, Canada.

[14] Beale R, Jackson T (1990) Neural Computing: An introduction, Institute of Physics Publishing.

[15] Carpenter GA, Grossberg S (1987) Invariant pattern recognition and recall by an attentive self-organizing art architecture in a nonstationary world, Proceedings of the IEEE First International Conference on Neural Networks, pages II-737-II-745.

[16] Georgiopoulos M, Heileman GL, Huang, J (1990) Convergence properties of learning in ART1, Neural Computation, 2(4) pp 502-509.

[17] Moore B (1988) ART and Pattern Clustering, Proceedings of the 1988 Connectionist Models Summer School, pp 174-183.

[18] Massey L (2003) Using ART1 Neural Networks to Determine Clustering Tendency, In Lotfi A, Garibaldi JM (Eds.), Applications and Science in Soft Computing, Springer-Verlag, pp 17-22.

[19] Ishihara S, Ishihara K, Nagamachi M, Matsubara Y (1995) arboART: ART based hierarchical clustering and its application to questionnaire data analysis, Proc. IEEE International Conference on Neural Networks, 1995, Vol 1, pp 532 -537.

[20] Bartfai G, White R (1997) Adaptive Resonance Theory-based Modular Networks for Incremental Learning of Hierarchical Clusterings, Connection Science, Vol. 9, No. 1, pp 87-112.

[21] Lavoie P, Crespo J-P, Savaria Y (1999) Generalization, discrimination, and multiple categorization using adaptive resonance theory, IEEE Transactions on Neural Networks, v.10 no.4, pp 757-67.

[22] Burke L (1995) Conscientious neural nets for tour construction in the traveling salesman problem: The vigilant net, Computer and Operational Research, vol. 23, no. 2, pp. 121-129.

[23] Bartfai G (1996) An ART-based Modular Architecture for Learning Hierarchical Clusterings, Neurocomputing, 13, pp 31-45.

[24] Larsen B, Aone C (1999) Fast and effective text mining using linear-time document clustering, Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp 16 – 22.

[25] Massey L (2005) An Experimental Methodology for Text Clustering, Proceedings of 2005 IASTED International Conference on Computational Intelligence (CI 2005), Calgary, Canada.

[26] Sebastiani F (2002) Machine learning in automated text categorization, ACM Computing Surveys, 34(1), pp1–47.

[27] VanRijsbergen CJ (1979) Information Retrieval, London: Butterworths.

[28] Larkey LS, Croft WB (1996) Combining classifiers in text categorization, Proc. of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval, Zurich, pp 289-297.

[29] Weigend AS (1999) Wiener ED, Pedersen JO (1999) Exploiting hierarchy in text categorization, Information Retrieval 1(3), pp 193-216.

[30] D'Alessio S, Murray M. Schiaffino R, Kershenbaum A (1998) Category levels in hierarchical text categorization, Proceedings of EMNLP-3, 3rd Conference on Empirical Methods in Natural Language Processing.

[31] Hotho A, Staab S, Stumme G (2003) Wordnet improves Text Document Clustering. Proc. of the Semantic Web Workshop of the 26th Annual International ACM SIGIR Conference, Toronto, Canada.

Fig. 1 ART1 Architecture

Fig. 2 The output neurons of two independent ART1 networks are used to create levels of a hierarchy. Since $\rho_0 < \rho_1$, the ART network functioning at ρ_0 distinguishes more general structure and thus its output is used on a higher level of the hierarchy

Fig. 3 Two ART1 networks in series. Prototypes from the lower network serve as input for the next network. Vigilance ρ_0 of the first level network must be large enough to cause the creation of multiple specific categories which can be amalgamated by the network at the next level

Fig. 4 F_1 quality for k-means and for HAC Ward's method

Fig. 5 Hierarchy visualization in a Web browser

Fig. 6 The cross-correlation matrix between clusters (the rows) and topics assigned by human classifiers (columns). The first row contains the topic numbers and the first column contains the cluster numbers (in bold)

Table 1 - Quality and number of clusters at each level for independent ART networks.

Table 2 - Sub-clustering for independent ART networks.

Table 3 - Characteristics of the hierarchy obtained with ART networks in series.

Table 4 - The most common words in reuter with cut-off frequency of 77 for dimensional reduction.

Table 5 – Sub-topics from class "acquisition" formed by ART.