An Integrated Cluster Analysis and Validity Test Platform for the Compression based Clustering Approach

Alexandra CERNIAN*, Dorin CARSTOIU, Adriana OLTEANU, Valentin SGARCIU

University Politehnica of Bucharest , 313 Splaiul Independentei, Bucharest, Romania.
Alexandra.cernian@aii.pub.ro

* corresponding author

Abstract: This paper focuses on the compression based clustering and aims to determine the most suitable combinations of algorithms for different clustering contexts (text, heterogeneous data, Web pages, metadata and so on) and establish whether using compression with traditional clustering methods leads to better performance. In this context, we propose an integrated cluster analysis test platform, called EasyClustering, which incorporates two subsystems: a clustering component and a cluster validity expert system, which automatically determines the quality of a clustering solution by computing the FScore value. The experimental results are focused on two main directions: determining the best approach for compression based clustering in terms of context, compression algorithms and clustering algorithms, and validating the functionality of the cluster analysis expert system for determining the quality of the clustering solutions. After conducting a set of 324 clustering tests, we concluded that compressing the input when using traditional clustering methods increases the quality of the clustering solutions, leading to results comparable to the NCD and the cluster analysis expert system proved 100% its accuracy so far, so we estimate that, even if some slight deviation should occur, it will be minimal.

Keywords: clustering, compression, cluster analysis, FScore, expert system.

1. Introduction

Clustering is an extremely powerful tool used for identifying patterns and grouping in datasets, based on the similarity between elements (Murty et. al., 1999). It is considered an unsupervised process (Charu and Chandan, 2013), since there is no predefined structure of the data. Clustering is applicable in many domains, ranging from biology and medicine to finance and marketing. It is used in fields such as data mining, pattern recognition, information retrieval, image analysis, market analysis, statistical data analysis and so on.

This paper presents the design, implementation and evaluation of a cluster analysis expert system, called EasyClustering, developed in order to assess the performance of different compression based clustering approaches and automatically computes the quality of the solutions. The system has 2 main integrated components:

1. A clustering component (Cernian et. al., 2011), with 3 compression algorithms (ZIP, bzip2 and GZIP), 4 distance metrics (NCD, Jaro, Jaccard and Levenstein) and 3 clustering algorithms (UPGMA, MQTC and k-means).
2. A cluster analysis expert system, which performs an automatic evaluation of the quality of the clustering results, using one of the most representative quality measures - the FScore (van Rijsbergen, 1976).

The research conducted with the EasyClustering platform has the following objectives:

1. To establish which is the most appropriate clustering context for using the compression based approach
2. To facilitate a comparative analysis of the clustering results produced by various combinations of compression algorithms, distance metrics and clustering algorithms
3. To evaluate the benefits of the compression based clustering approach
4. To provide an expert system component to automatically assess the quality of the clustering solutions
5. To investigate if traditional clustering methods have improved performance when the input is compressed

The rest of the paper is structured as follows: Section 2 presents the theoretical background and some related work, Section 3 describes the EasyClustering platform and the methodology for using the platform, Section 4 presents some experimental results for validating the capabilities of this integrated system, and Section 5 draws the conclusions for this work.

2. Cluster Validity State of the Art

At present, there are several clustering platforms available, such as:

- Cluster 3.0 (de Hoon et. al., 2004), which is dedicated to genomic datasets hierarchical clustering
- CompLearn Toolkit (Cilibrasi, 2003), dedicated to clustering by compression. It also uses the NCD distance metric.
- RapidMiner (RapidMiner, 2011), Weka (Weka, 2011), 2 data mining platforms that integrate some clustering components.
- ClusTIO (ClusTIO, 2009) is a Java command-line clustering tool implementing several clustering algorithms (such as UPGMA and VCQM)
- CVAP (Wang, 2009), which is a cluster validity and analysis tool, including several validity indices.

The drawback is that there has been little attention given to an automatic cluster analysis approach. Most current clustering platforms stop at a visual representation of the clusters produced by various clustering algorithms. CVAP is the only cluster analysis and validation platform, but it uses different validity indices than those proposed in this paper. In this context, we propose the design and implementation of an integrated cluster analysis and validity test platform, called EasyClustering.

2.1 Cluster validity indices

The main aspect concerning cluster validity is evaluating the correctitude of the clustering. The purpose of clustering methods is to discover significant groups in a dataset. These groups are called clusters. Generally speaking, clustering algorithms should search for clusters whose members are close to each other (in other words have a high degree of similarity), but well separated from the members of the other clusters. The procedure of evaluating the quality of the results produced by clustering algorithm is known as *cluster validity*.

There are three main approaches for the cluster validity process (Fowlkes and Mallows, 1983): external criteria, internal criteria and relative criteria. *External criteria*: the results produced by the clustering algorithm are compared against a pre-specified structure. *Internal criteria*: the results of a clustering algorithm are evaluated based on the distance matrices computed in the clustering process. *Relative criteria*: a clustering solution is compared against other clustering structures, generated by the same algorithm, but with different parameter values. There are two main criteria proposed for evaluating the clustering solutions and selecting the optimal clustering structure (Batistakis et.al., 2002):

- *Compactness*: the members of each cluster should be as similar as possible. A common measure of compactness is variance, which should be minimized.
- *Separation:* the clusters should be as clearly separated as possible. The distance between clusters can be measured using one of the following approaches: single linkage, complete linkage and comparison of centroids.

Throughout this paper, we will be interested in the FScore (van Rijsbergen, 1979) quality measure. The FScore was implemented in the cluster validity component in order to automatically compute the quality of the clustering solutions. This is how this FScore is defined: let us consider L_r a class of size n_r and Si a cluster of size n_i, resulted after a clustering process, and assume that n_{ri} items in the cluster S_i belong to L_r. In this case, the FScore is computed as (Rijsbergen, 1979):

$$F(L_r,S_i)=\frac{2*R(L_r,S_i)*P(L_r,S_i)}{R(L_r,S_i)+P(L_r,S_i)} \quad (1)$$

where $R(L_r,S_i)=\frac{n_{ri}}{n_r}$ is the recall value, which is a measure of the cluster containing all the correct items, and $P(L_r,S_i)=\frac{n_{ri}}{n_i}$ is the precision value, measuring the number of items in a cluster that truly belong there.

Section 3 presents the architecture and design of the cluster analysis expert system.

3. The Architecture of the Cluster Analysis and Validity Expert System

Figure 1 presents the architecture of the expert system component.

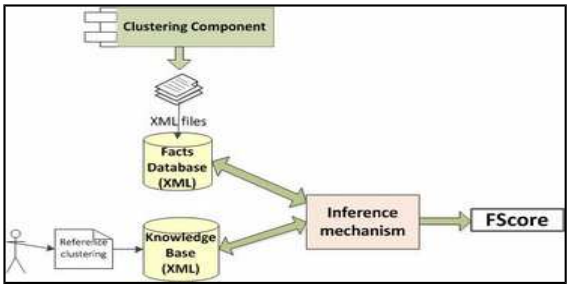


Figure 1. The architecture of the EasyClustering cluster analysis expert system

Facts Database. This component of the expert system contains the clustering solutions generated by the clustering component. When the clustering component produces a new set of results, they are saved in a predefined XML format (XML Standard, 2011) and saved in the application folder. This is what such an XML file looks like:

```
Facts Database
<?xml version=="1.0?>
<clustering>
<cluster number="1"/>
<document><id>12/id></document>
<document><id>13/id></document>
</cluster><cluster number="2"/>
<document><id>0</id></document>
<document><id>1</id></document>
<document><id>2</id></document>
<document><id>4</id></document>
<document><id>5</id></document>
<document><id>6</id></document>
<document><id>7</id></document>
<document><id>8</id></document>
<document><id>9</id></document>
<document><id>10</id></document>
<document><id>11</id></document>
<document><id>14</id></document>
<document><id>16</id></document>
</cluster>
<cluster number="3"/>
<document><id>3</id></document>
<document><id>15</id></document>
</cluster>
</clustering>
```

Knowledge base. This component of the expert system stores the correct clustering **solutions,** and it is used as a reference when computing the FScore for a dataset. The knowledge base is made up of XML files, having the same format as the files produced by the clustering component.

So, when a user decides upon a dataset to be clustered, he must first use the clustering component in order to obtain an automatic solution. When the clustering component has generated its results, the user can evaluate the quality of the solution obtained by using the cluster validity expert system of the EasyClustering platform. He will upload the XML file provided by the clustering component, which is stored in the facts database, as well the reference solution (also an XML file), which will be stored in the knowledge base. Afterward, the 2 XML files will be parsed, in order to compute the FScore of the classification produced by the clustering component.

The inference mechanism. This component which takes as input the 2 XML files described above and computes the FScore of the solution produced by the EasyClustering clustering component.

```
Knowledge Base
<?xml version=="1.0?>
<clustering>
<cluster number="1"/>
<document><id>12</id></document>
<document><id>13</id></document>
<document><id>5</id></document>
<document><id>6</id></document>
<document><id>7</id></document>
</cluster>
<cluster number="2"/>
<document><id>0</id></document>
<document><id>1</id></document>
<document><id>2</id></document>
<document><id>4</id></document>
<document><id>8</id></document>
<document><id>9</id></document>
<document><id>10</id></document>
<document><id>11</id></document>
</cluster>
<cluster number="3"/>
<document><id>3/id></document>
<document><id>15/id></document>
<document><id>14</id></document>
<document><id>16</id></document>
</cluster>
</clustering>
```

The FScore formula is also stored in the knowledge base, so the expert system can be easily extended by introducing new quality measures in the knowledge base, together with the parsing mechanism required for their calculations.

Besides the formula retrieved from the knowledge base, the inference mechanism also contains a procedural component, which provides the algorithm used to parse the 2 XML files in order to obtain the value of the FScore.

The platform was implemented in Java and therefore it is fully portable.

3.1 The UML model of the expert system

In order to provide a better understanding of the capabilities of the EasyClustering cluster analysis expert system, we will present a part of its UML model, namely the use case diagram and the activity diagram, which presents the flow of actions in a more accurate manner.

Figure 2 illustrates the UML use case diagram of the expert system.

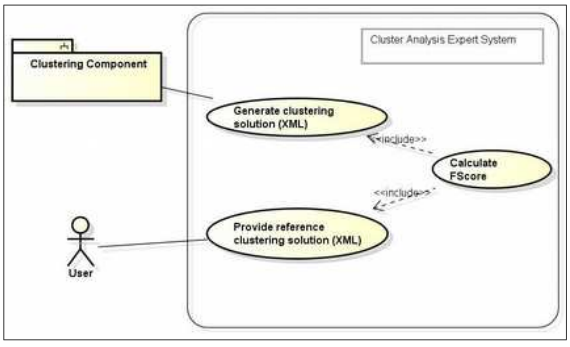


Figure 2. The use case diagram the EasyClustering cluster validity expert system

There are 2 actors interacting with the system: the clustering subsystem, which provides an automatically generated clustering solution, and the user, who is in charge with storing the correct reference clustering in the expert system knowledge base. Based on these 2 inputs, the inference mechanism will compute the value of the FScore, in order to evaluate the quality of the solution resulted from the clustering component.

Figure 3 represents the UML activity diagram for the cluster validity expert system.

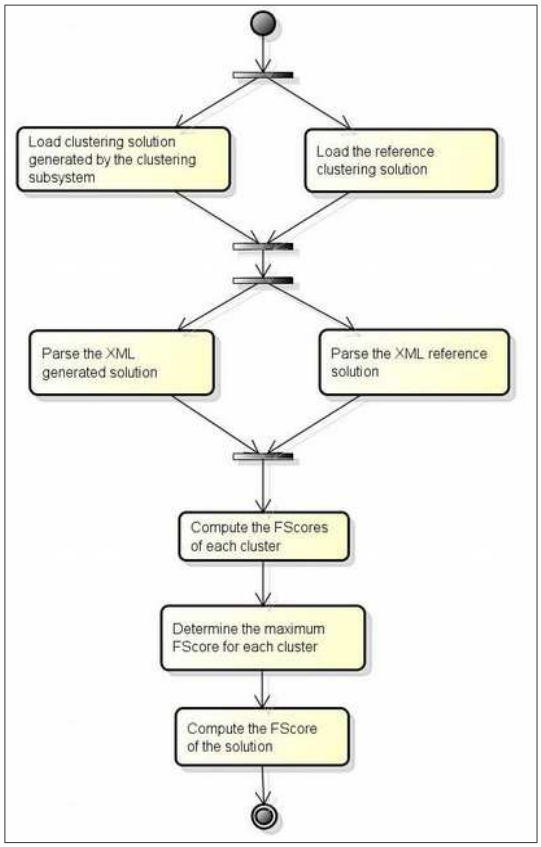


Figure 3. The activity diagram the EasyClustering cluster validity expert system

The parsing phase consists of a procedural approach and has the following steps:

1. Extract the clusters of the reference solution and the id's of the documents in each cluster
2. Compute n_r for each cluster in the reference solution.
3. For each <cluster> element of the generated solution
   - 3.1 Determine the id's of documents which belong to the cluster
   - 3.2 Compute n_i.
   - 3.3 Compute n_{ri} for each pair of clusters(current cluster extracted from the generated solution – each cluster in the reference solution)
   - 3.4 Compute the FScore of the cluster.
   - 3.5 Retain the maximum value of the FScore for the cluster.
4. Compute the FScore of the solution.

The main steps involved in the compression based cluster analysis and evaluation process using the EasyClustering platform are the following:

1. **Choose dataset.** The first step in any clustering process is the selection of a proper dataset, which will be submitted as input for the clustering platform. In our case, there are no restrictions regarding the input dataset. It can contain homogeneous or heterogeneous data, in whatever format the user wants to test.
2. **Clean dataset**. There are situations when the dataset will need to go through a cleaning phase before being submitted to the clustering component. This step is not mandatory and it comprises the following options: stopwords removal and stemming algorithm.
3. **Choose compression algorithm.** The user can select the compression algorithm to be used. The following options are available in the EasyClustering platform: ZIP, bzip2 and GZIP. If the distance metric used to compute the distance matrix is the NCD, then this step is mandatory. Otherwise, the user is not forced to select a compressor and the files will be used in their original format.
4. **Choose distance metric.** The user can select the distance metric used to compute the similarity matrix. The following options are available in our platform: NCD, Jaro, Jaccard and Levenstein. New distance metrics are easy to include in the EasyClustering platform.
5. **Choose clustering algorithm**. Once the distance matrix has been created, it will be interpreted by a clustering algorithm, which will generate the hierarchy of files. There are 3 clustering algorithms available in our system: K-Means, UPGMA and MQTC. If the user chooses the K-Means option, he will also be invited to submit the number of clusters, K.
6. **Generate the clustering hierarchy using the clustering component.** All the choices from the previous steps are combined, in

order to generate the clustering solution. The hierarchy will be displayed in a graphical format, as well as saved in XML format in the application folder. This folder is the facts database of the cluster validity expert system.

7. **Provide the reference clustering in XML format.** The user must provide the correct clustering solution for each specific dataset. These files have a predefined XML format, compatible with the format used at step 5, and they form the knowledge base of the expert system.
8. **Compute FScore**. The inference mechanism parses the 2 XML files and the expert system computes the FScore of the clustering solution, using a procedural algorithm.

Figure 4 depicts the methodology for using the EasyClustering platform.

4. Experimental Results

The experimental validation of the platform focused on the following objectives:

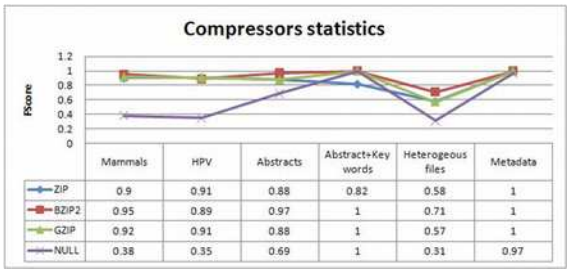
- To assess the overall quality of the clustering solutions produced when integrating compression algorithms in the process.
- To investigate if the performance of traditional clustering methods is improved when the input is compressed
- To assess the performance of the FScore based cluster validity expert system

In order to address our objectives, we have conducted 324 tests, using 9 datasets:

- 32 mammals genomes
- 30 human papillomavirus genomes
- 15 text files
- Handwritten text
- Heterogeneous files (Carstoiu et. al., 2009)
- Metadata associated to heterogeneous files
- Web pages - snippets
- Web pages (Cernian et. al., 2011)
- Processed Web pages (HTML tags removal, stopwords removal, stemming) (Cernian et. al., 2011)

First of all, let us analyse the benefits of using compression with traditional clustering approach and compare these results with the clustering by compression technique based on the NCD distance metric.

Figure 5 presents the general statistics of the best results obtained for 6 datasets, analysed from the perspective of compression algorithms. We also took into account the no compression alternative, in order to get an objective comparative view of the benefits or disadvantages of the compression based approach. The clustering results were obtained with the UPGMA algorithm.



	Mammals	HPV	Abstracts	Abstract+Key words	Heterogeous files	Metadata
ZIP	0.9	0.91	0.88	0.82	0.58	1
BZIP2	0.95	0.89	0.97	1	0.71	1
GZIP	0.92	0.91	0.88	1	0.57	1
NULL	0.38	0.35	0.69	1	0.31	0.97

Figure 5. Overall statistics from the compressors perspective

For all 6 datasets, the results produced by the 3 compression algorithms, ZIP, BZIP2 and GZIP are relatively close, especially for clustering gene expressions. The no compression

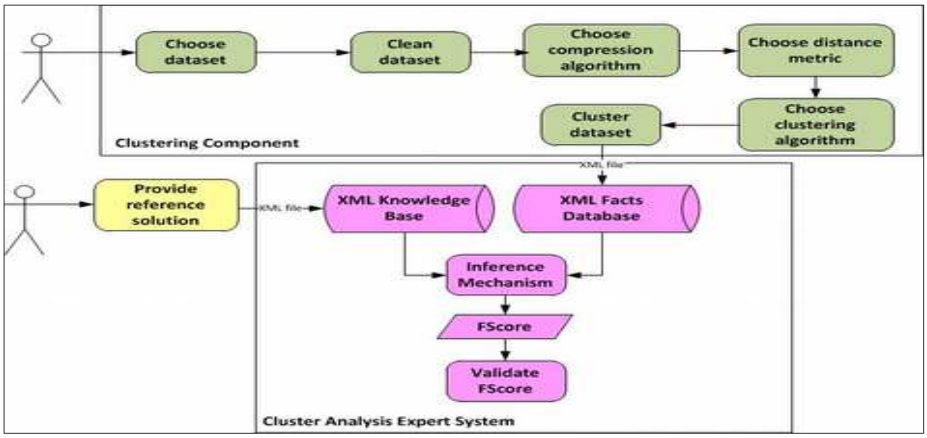


Figure 4. The methodology of using the EasyClustering cluster analysis and validity platform

approach led to poor results when clustering the mammals and HPV genomes, as it can also be noticed in Figure 5.

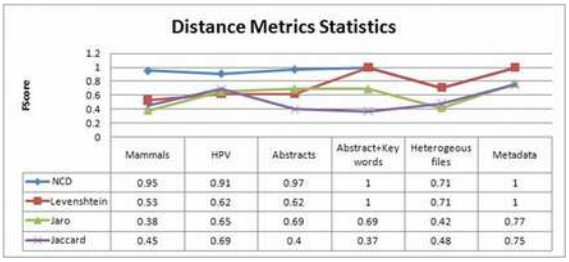
The BZIP2 and GZIP lines are almost parallel. For the first 3 datasets, the ZIP line is almost identical to the GZIP line, but we notice a significant drop when clustering the scientific abstracts. An interesting aspect is the ZIP line reaches the maximum FScore value of 1 for clustering heterogeneous files based on associated metadata. Why this difference of quality between clustering scientific abstracts with increased keywords' weight and clustering metadata files with increased keywords' weight? The reason is the size of the files. The metadata files are significantly smaller than the scientific abstracts and the ZIP compressor produces better results for small files.

An overall analysis leads us to the conclusion that the best results were obtained for the BZIP2 compressor, followed by GZIP and by ZIP.

The no compression approach produced very good results for clustering text files with increased keywords' weight, thus files where the most relevant pieces of information have been stressed.

As a conclusion, the results confirmed that compressing the input leads to improved results when using traditional clustering algorithms, such as UPGMA.

Figure 6 presents the general statistics of the best results obtained for each of the 6 datasets, analysed from the perspective of distance metrics.



	Mammals	HPV	Abstracts	Abstract+Key words	Heterogeous files	Metadata
NCD	0.95	0.91	0.97	1	0.71	1
Levenshtein	0.53	0.62	0.62	1	0.71	1
Jaro	0.38	0.65	0.69	0.69	0.42	0.77
Jaccard	0.45	0.69	0.4	0.37	0.48	0.75

Figure 6. Overall statistics from the distance metrics perspective

Judging from the distance metrics perspective, it is obvious that the best results were produced by the Normalized Compression Distance (NCD). Almost all peak values obtained for the NCD are close to, or even reach, 1, which means a perfect match with the reference solution.

For the first 3 datasets, we notice that the NCD line is significantly above the other 3 lines. For the HPV gene expressions, the Levenshtein, Jaro and Jaccard distance metrics produce very similar results, with FScores around the value of 0.65. For the mammalian genomes, these 3 metrics produce lower values, with an average of 0.45. However, the top value produced by the NCD is better than the top value of the HPV dataset (0.95 vs. 0.91). To conclude, for the gene expression datasets, the only accurate results were obtained when using the NCD, which confirms the results presented in [4]. The other 3 metrics produced average to poor results (with or without compressing the datasets).

For the text clustering experiments, the best results were also produced by the NCD. When clustering the scientific abstracts (dataset 3), the other 3 metrics did not produce good results, although Jaccard is one of the distance metrics commonly used for clustering text files. On the other hand, the Levenshtein metric produced a very good outcome for clustering the abstracts with an increased weight for the keywords. The FScore of 1 was obtained when the dataset had not been compressed.

Heterogeneous files were significantly better clustered based on associated metadata. Once again, the Levenshtein metric also produced a maximum FScore. The results for this particular metric did not change significantly when input files were compressed.

A total number of 324 tests have been conducted and analysed (9 datasets x 3 compressors x 4 distance metrics x 3 clustering algorithms). The average quality of the results per dataset ranged from an FScore of 0.49 to 0.97.

Figure 7 presents a statistical analysis of the best results obtained for the 9 datasets mentioned above.

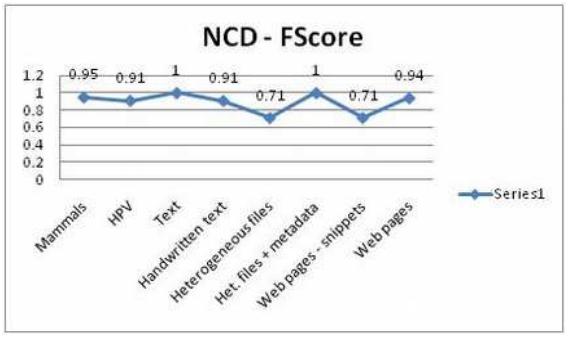


Figure 7. The highest FScore values

Overall, the best solutions have been produced by the NCD distance metric combined with the UPGMA clustering algorithm. The results obtained when using MQTC were very similar to those produced by UPGMA, but the algorithm is very time-consuming compared to UPGMA. K-means lead us to very good results when the number of clusters, K, was accurately provided. Moreover, we concluded that clustering solutions are improved when compressing the input, when using other distance metrics than NCD. The results were highly comparable to those obtained with the NCD for all datasets except genomes.

The clustering subsystem was created a few months before the FScore based cluster validity expert system. After implementing the cluster validity system, our main interest was to check the accuracy of its results. Therefore, we started our experiments with 2 datasets, of 15 and 17 elements, and we pursued the following procedure:

- We produced a clustering solution with the EasyClustering platform, using the following combination of algorithms: ZIP + NCD + K-Means. The results were exported in the XML format specific to the cluster validity expert system and stored in the application folder (the facts database of the expert system).
- We manually produced the clustering structure used as reference, in XML format, which we stored in the expert system knowledge base.
- We loaded the 2 files into the expert system and launched the FScore computation.
- We manually computed the FScore
- We compared the 2 Fscore values, in order to evaluate the accuracy of the cluster validity expert system.

Table 1 contains the FScore values generated by the expert system for the 2 datasets:

Table 1. FScore values produced by the EasyClustering cluster validity expert system

	Dataset 1 (17 items)	Dataset 2 (15 items)
FScore	0.68	0.50

Table 2 contains the FScore values manually computed for the 2 datasets:

Table 2. FScore values manually computed

	Dataset 1 (17 items)	Dataset 2 (15 items)
FScore	0.68	0.50

As noticed from tables 1 and 2, the cluster validity expert system had a 100% precision for these 2 datasets. Thus, we can estimate that, even if a small deviation should occur, it will be minimal.

5. Conclusion

Life is all about choice. We make choices on a daily basis, in almost every aspect of our lives. Our ability as humans to accumulate and process information relies on our ability to structure the information that we receive and find logical connections between the information that we have. Philosophers, starting with Aristotle, have been preoccupied with discovering the existence of things. Once established that something exists, the next step is to describe how it fits among other things whose existence has been proven. Intelligent applications follow the same principles and achieve the same results using two categories of algorithms—*clustering* and *classification*. For classification**,** the task is to learn to assign objects to predefined classes, while for clustering**,** no predefined structure is required. The task is to learn a classification from the data. In other words, clustering is an unsupervised learning process (Marmanis and Babenko, 2009), while classification is considered to be a supervised learning process.

In this context, we propose the work presented in this paper, namely the EasyCLustering cluster analysis expert system, which was conceived in order to evaluate the benefits of using compression in the clustering process. It is made up of 2 components: the clustering component and the FScore based cluster analysis expert system, which automatically determines the quality of the clustering solution provided by the clustering component. The connection between the two components is made with XML technological space.

The originality aspects of the paper consist in the following contributions: the design and implementation of the EasyClustering cluster analysis expert system, investigating if traditional clustering methods have improved performance when the input is compressed, determining the most appropriate clustering context for using the compression based approach and integrating the FScore as a quality index in order to automatically compute the quality of the clustering solutions generated by the platform.

The experimental results focused on two main directions: assessing the benefits of integrating compression into the clustering process and validating the functionality of the cluster analysis expert system. After conducting a set of 324 clustering tests, we drew the conclusion that integrating compression algorithms in the process leads to good results, the FScore values obtained reaching a top value of 0.97. Moreover, the cluster analysis expert system proved 100% accuracy so far, so we estimate that, even if some slight deviation should occur, it will be minimal.

As future work, we plan extend the platform into a distributed environment in order to improve the speed performance (Mocanu et. Al., 2014). More precisely, we plan to design and implement the integration of EasyClustering platform with Hadoop and test the clustering performance obtained with MapReduce.

Acknowledgments

The work has been partly funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Ministry of European Funds through the Financial Agreement POSDRU/159/1.5/S/132395.

REFERENCES

1. BOOCH, G., I. JACOBSON, J. RUMBAUGH, **OMG Unified Modelling Language Specification**, First Edition: 2010. http://www.omg.org/spec/UML/2.3/
2. BZIP2 home page: http://bzip.org/, last accessed 23.06.2014.
3. CILIBRASI**,** R., **The CompLearn Toolkit**, http://www.complearn.org/, 2003.
4. CILIBRASI R, VITÁNYI, PAUL M.B., **Clustering by Compression**, IEEE Trans. on Info. Th., vol. 51, 2005, pp. 1523-1545.
5. CLUSTIO, http://www.softpedia.com/get/Science-CAD/ClusTIO.shtml, 2009.
6. DE HOON, M. J. L., S. IMOTO, J. NOLAN, S. MIYANO, **Open Source Clustering Software**, Bioinformatics, vol. 20(9), 2004, pp. 1453-1454.
7. CHARU C. A., C. K. REDDY, **Data Clustering: Algorithms and Applications**, CRC Press, 2013.
8. MARMANIS, H, D. BABENKO, **Algorithms of the Intelligent Web**, Manning Publications, 2009.
9. MURTY, M., A. JAIN, P. FLYN, **Data Clustering: A Review**, ACM Computing Surveys, vol. 31(3), 1999.
10. MILLIGAN, G. W. **Clustering Validation: Results and Implications for Applied Analyses**, World Scientific Publ., 1996.
11. RAPIDMINER http://rapid-i.com/content/view/181/196/., accessed 10.06.14.
12. WANG, K., **CVAP: Cluster Validity Analysis Platform** (cluster analysis and validation tool), at: http://www.mathworks.com/matlabcentral/fileexchange/14620-cvap-cluster-validity-analysis-platform-cluster-analysis-and-validation-tool, 2009.
13. HALL, M., E. FRANK, G. HOLMES, B. PFAHRINGER, P. REUTEMANN, I. H. WITTEN, **The WEKA Data Mining Software: An Update**; SIGKDD Explorations, vol. 11(1). 2009.
14. CERNIAN, A., SGARCIU, V., CARSTOIU, D., **Experimental Validation of the Clustering by Compression Technique**, U. P. B. Scientific Bulletin, Series C, vol. 73(3), 2011, pp. 61-74.
15. VAN RIJSBERGEN, C. J., **Information Retrieval**, 2nd ed., Butterworth, 1979.
16. CERNIAN A., D. CARSTOIU, **Clustering Heterogeneous Data using Clustering by Compression**, Proc. 13th WSEAS Intl. Conf. on Computer*,* 2009.
17. CARSTOIU, D., A. CERNIAN, V. SGARCIU, A. OLTEANU, **A New Method for Clustering Heterogeneous Data: Clustering by Compression**, Journal WSEAS Transactions on Computers*, v*ol. 8(9), Sept. 2009, pp. 1461-1470.
18. CERNIAN, A., D. CARSTOIU, A. OLTEANU, **Clustering Heterogeneous Web Data using Clustering by Compression. Cluster Validity**, 13th Intl. Symp. on Symbolic and Numeric Algorithms for Scientific Computing, 2011.
19. MOCANU, S., R. DIN, D. SARU, C. POPA, **Using Graphics Processing Units for Accelerated Information Retrieval**, Studies in Informatics and Control, vol. 23 (3), 2014, pp. 249-256.