Text Mining Grid Services for Multiple Environments

Antonio Anddre Serpa, Valeriana G. Roncero, Myrian C. A. Costa, Nelson F.F. Ebecken

COPPE/Federal University of Rio de Janeiro,
P.O.Box 68516, 21945-970 Rio de Janeiro RJ, Brazil
Tel: (+55) 21 25628081, Fax: (+55) 21 25628080
serpa@nacad.ufrj.br, valery@nacad.ufrj.br, myrian@nacad.ufrj.br, nelson@ntt.ufrj.br

Abstract. The objective of this paper is to describe the implementation of text mining grid services for Aîuri Project, which is a framework that includes a friendly user interface, data and text mining tasks, database access and a visualization tool integrated with various grid environments. The focus is the development and test of components for analysis and evaluation of unstructured data into distinct grid environments. These components will be grid services for text mining processes using several approaches of execution, depending on which grid environment the user choose to submit. All components are open source and are freely available to the scientific community, providing access to existing services as well as encouraging the addition of new ones.

Keywords: Text Mining, Categorization, Grid Computing and Portal.

1. Introduction

Due to the continuous growth of the volume of available electronic data, automatic knowledge discovery techniques become necessary in order to manipulate huge amounts of data. Huge amounts of numerical data and countless pages of texts are produced every day, in the academic or enterprise fields, documenting projects, actions or ideas. All the knowledge expressed in structured or unstructured form represents the most important property of an institution, either competitive advantage for companies or the availability of concepts and ideas for the academia. The techniques of text mining aims at extracting implicit knowledge in a collection of texts and documents [1].

Nowadays, the advances in education and research in the areas of text mining are leading to a torrent of new algorithms and methodologies for solving complex engineering and advanced sciences problems. Teaching those new algorithms and methods becomes itself a big challenge, if it is supposed that the use of programs with a friendly user interface and efficient visualization tools is necessary, even though this is not the main focus of the work and sometimes it is not present. Such difficulties can be minimized with the utilization of a well-defined environment that contains the previously mentioned facilities, leading to time savings on development, as well as keeping the focus on development and test of algorithms.

Grid computing is an infrastructure that integrates distributed and heterogeneous hardware and software resources, providing a virtual platform for computation and data management [2]. The growth of the environments of cooperative research enables the collaboration of researchers from geographically scattered research centers. This scenario requires new dynamics in the research areas, allowing the development of new methodologies that integrate several development phases of new applications.

However, the benefits that the use of computational grids brings can not be fully explored if they can not be easily accessed by an ordinary user. The user needs a friendly interface to interact with the grid environment. The contribution of this work is the construction of a friendly tool, where the users can submit their tasks into three distinct environments, so that they can select the most suitable one to his task. This access is transparent to the user [16]. To test the tool there were used two known algorithms of text mining: Naïve Bayes and Linear Score, using the concepts of grid services.

2. Description of the Aîuri Project

The objective of Aîuri Project is the creation of a framework that aggregates a friendly user interface and the tasks of text mining, using one or more grid environments, in way to make easy its use for the researches. The focus of the project is the development of a high performance academic cooperative environment, which will be used for education and research in the areas of computational intelligence, analysis, evaluation and visualization of data via grid services that encapsulate the algorithms of data and text mining process. The software integrated in this environment is open source and available for the community, which will be capable of accessing built-in services and/or add new ones.

Grid computing infrastructure aggregates the collaboration feature, allowing the utilization and/or incorporation of new strategies for the research of new algorithms and different approaches for the solution of advanced engineering and science problems.

This paper describes the implementation of a text mining grid service for the Aîuri Project.

3. Text mining tasks on the Aîuri environment

Text Mining, also known as Knowledge Discovery form textual databases [3], refers to the non trivial extraction of implicit, previously unknown, and potentially useful information from large amounts of textual data, such as documents and unstructured data. Text Mining describes an assembly of processes with algorithms and efficient techniques that allow the manipulation of texts. Different algorithms can be used depending on the discovery goal.

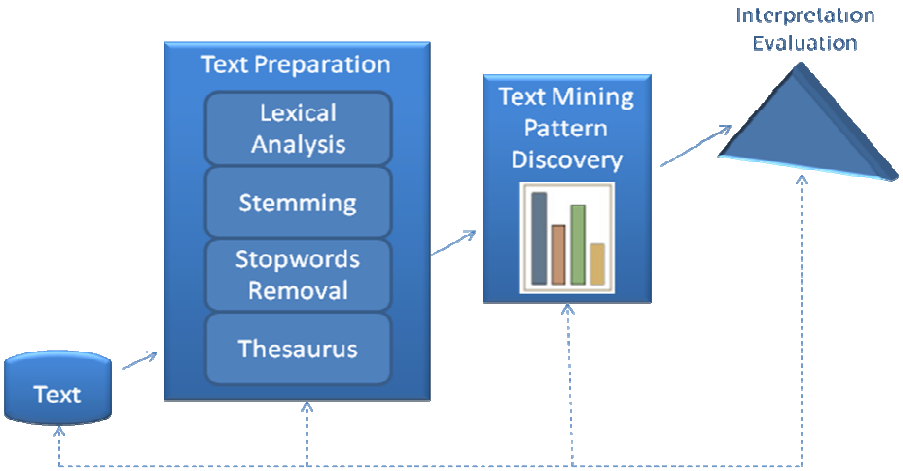

Figure1: Scheme of Text Mining Process, showing its main phases

The main phases of text mining process are text preparation, text mining pattern discovery and post-processing or interpretation and evaluation, as shown in figure 1.

3.1 Text Preparation

The text preparation phase consists of the selection of terms that better describe document contents, neglecting any unimportant information. This type of selection improves performance and classification effectiveness. The activities that must be performed are: lexical analysis that identifies each term as a character sequence; morphological analysis or stemming that reduces each term to its radical, made by a stemmer; stop words removal that consists of removing terms with no special meaning from the texts, like prepositions, articles and conjunctions; utilization of a thesaurus in order to replace different terms with a key term that has the semantic meaning.

The use of these techniques results in a collection of representations of words of a document, which is mapped into a term-document table.

3.2 Text Mining Pattern Discovery

Among the diverse approaches of analysis for the extraction of knowledge, categorization tasks are intended to automatically classify documents related to a collection of previously defined categories [4]. This task can be used, for example, to insert a new document in a collection divided into categories. In order to do that, the categories have to be represented by terms or a set of terms that bear the meaning of the category concept. The categorization technique is defined as the process of finding a model that describes a category. Given a collection of labeled records, each of them containing a set of features and the category, the model for a category is a function of the values of the features. Usually, the data set is divided into training and test sets, where the training set is used to build the model and test set is used to validate it and determine its accuracy. The goal of this process is to assign categories to previously unseen records as accurately as possible.

3.3 Post-Processing

The post-processing task consists of the validation, visualization and evaluation of the obtained patterns from the expert point of view. The evaluation accuracy and the visualization tools are especially important in order to achieve the most useful and relevant conclusions. This phase is not treated in the present paper, because post-processing is subject of research and development in future projects.

4. The Aîuri Portal

The Aîuri portal is a web interface that allows users to request execution of text mining tasks, in three distinct environments. Because of the different features of each type of task, the application behavior is different, depending on how the user chooses the task parameters. The access to the portal is controlled by a user/password authentication method. All the services are available to all users, shown in the table 1.

Table 1. Portal Functions

Function	Description
Upload Certificate	It carries out the load of the certificate.
Upload File	It carries out the load of the files of the user.
Training set XML	Generation of the file with the training set.
Test set XML	Generation of the file with the test set.
Make Stemmer	Generation of the file with stems.
Bayesian categorization	Naïve Bayes categorizer.
Linear categorization	Linear Score categorizer.

The portal encapsulates the structure of Aîuri project. The Aîuri project has three main components: a portal, which is the interface between user and the services, the grid services, which implement the tasks of text mining and maintain all the files uploaded to the environment.

Web-based Grid computing portals are effective tools for providing users with simple, intuitive interfaces for accessing grid information and its resources [5]. The software used to build grid portals interacts with the middleware running on the resources. The portal software must be compatible with common Web servers and browsers/clients. Grid portals make the distributed heterogeneous computing and data grid environments more accessible to grid users by using common Web and UI (Users Interfaces) conventions.

The processes performed by the portal are shown in the figure 2. The first step is the upload of the data necessary to perform the mining task. For the execution of a text mining task all the text files which are part of the knowledge base must be uploaded. Each user could have a private knowledge

base in an exclusive area. Once the files are uploaded, the next step is to create the training and test sets, by converting them to the XML format. These sets will be stored in the user area too. In the grid environment these uploaded and converted files can be stored in the environment.

After that, the user can choose the parameters of the text mining task, enabling the portal to generate an object with the states of the parameters. Among other parameters, the user can specify the name of the dictionary, the amount of words and the key term for the categorization task.

At this moment the portal queries the available resources for the grid execution and lets the user choose the appropriate resource. The grid information service provides the states and workload of the grid resources. For the submission of the text mining task for execution, the portal contacts the job management service of the grid and waits for the task results.

The results are sent back to the portal and visualized by the user.

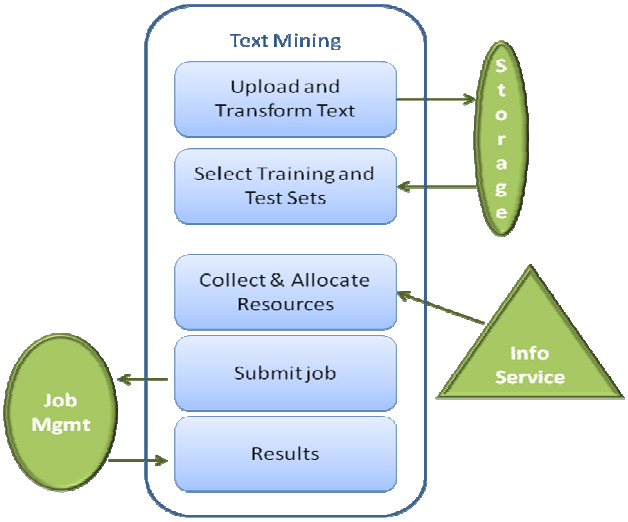

Figure 2 – Aîuri portal processes

4.1 Text Mining Grid Service

The first application implemented is text mining categorization. The application was developed in Java and is performed by the Naïve Bayes and Linear Score algorithms. The probabilistic Naïve Bayes (NB) classifier has been widely used with good performance for document classification. It is based on the Bayes' Theorem [6,7]. The basic idea is to join the key words in categories to estimate the probabilities of the categories of a new document. The algorithm computes the a posteriori probabilities of a document to belong to distinct classes and the attributes it to the class with larger a posteriori probability. The a posteriori probability is computed using the Bayes' rule and the test set is attributed to the class with the largest a posteriori probability. The naïve part of the NB algorithm is the independence assumption of the characteristics of the word, that is, it is assumed that the effect of the characteristics of the word of which conditional probability is associated to a category is independent of the characteristics of the other words of that category. Several experiments have been performed with the NB algorithm [8], presenting satisfactory results for pattern classification. However, other methods are also used to other application in the text mining area. The NB algorithm presents several advantages over other techniques. It is quite simple and easy to implement. Additionally, no learning process is required, because the probabilities are estimated based on the frequency of the terms. Moreover, the classification process is efficient, since the characteristics are independent of each other. On the other hand, the NB algorithm has some drawbacks. It requires many probabilities to be known *a priori*, and the computing cost grows linearly, depending on the quantity of existing words and characteristics.

The linear score classifier is based on linear methods, which are a classical approach for the resolution of predicting problems. The Bayesian method, used in this work, can be viewed as a special case of the linear method, but without problems with redundant attributes, since it performs better when the number of attributes is small, which is not the case when dictionaries with thousands of words are created. The linear score method [11] sets a positive score to the classes identified as positive and a

negative one to the classes identified as negative, such that for every word that appears in a document its corresponding weight is determined. These weights must be summed to compute the score of the document. An advantage of the linear approach is the simplicity of the construction of the model, provided that a set of significant terms of the document is chosen and the learning algorithm is capable of determining the weight of each term created.

5. Grid Environments

A grid is an internet-connected computing environment in which computing and data resources are geographically distributed over different administrative domains, often with separate policies for security and use of resources [9]. Two distinct computational grid environments are used in this work: the NACAD Grid, installed at the NACAD laboratory, and the EELA Grid.

5.1 NACAD Grid Environment

The NACAD Grid uses Globus GT4 [10] as grid middleware. In order to integrate the framework to the GT4 infrastructure, some entities were created to allow the integration of the system with this new environment, as illustrated in figure 4. GT4 is a grid middleware based on grid services. Grid services is a technology based on the concepts and technologies of grids and web services and can be defined as a web service that delivers a set of interfaces that follows specific conventions. It is a technology was originated from the necessity to integrate services through virtual, heterogeneous and dynamic organizations, composed of distinct resources, whether within the same organization or by resource sharing. The structure of the NACAD grid is shown in figure 3.

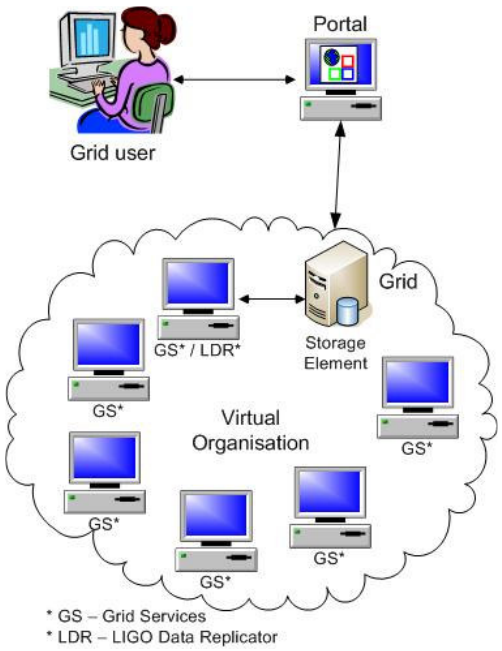

Figure 3 – Components of the Aîuri project

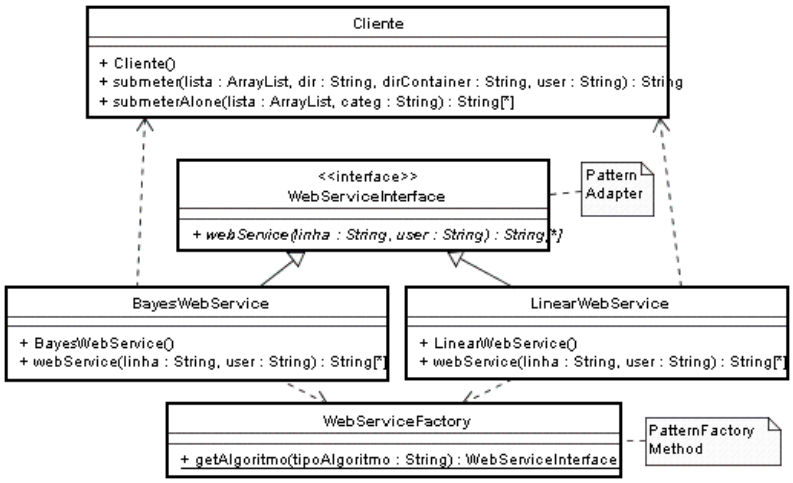

Figure 4 – Classes model for grid services

Notice that the design pattern Adapter is implemented at this class. The interface WebServiceInterface must be inherited by any class in which some algorithm is implemented and that needs to be called by means of grid services. The method webService(), implemented by means of the child class, encapsulates all the calls to the business class methods in which, in fact, the algorithms are implemented. To create the instances of those classes, it is used the Factory Method design pattern. One opted for the use of design patterns, since they are techniques that reduce the cost of evolution of the developed softwares.

5.2 EELA Grid Envrionment

The second integration was carried out within the EELA (E-Infrastructure Shared Between Europe and Latin America) Project. The objective of the EELA Project is to establish a human collaboration network to share an infrastructure to support test and development of advanced applications. EELA-2 is the second phase of EELA Project. The integration to this environment required the implementation of the following activities: (i) Use of AMGA (ARDA Metadata Grid Application): AMGA [12] is a metadata service for computational grids. It can be viewed as a data base access service for grid applications, which enables jobs running on the grid to access the data base, providing authentication, as well as a layer that hides from the user the technical details of distinct data bases, providing the user a unique method of access to all data bases in the environment. In fact, AMGA is a service that functions between the SGDB and the client application. AMGA is used to create a structure that validates the user at the moment he logs into the environment; (ii) Use of GSAF (Grid Storage Access Framework): GSAF [13] is a framework that encapsulates the access method to data by providing API's that provide access to AMGA, to the files catalogue and storage elements. The structure of AMGA and use of GSAF are shown in figures 5a and 5b; (iii) Job submission: to submit jobs to the EELA grid the API LCG (LHC Computing Grid) is used. This API provides a number of functionalities to access this environment. To submit jobs using this API a file that describes the properties of the jobs must be created. The Job Description Language [14] (JDL) is a language intended to create such description. The class implemented to perform the submission activities is an example of a JDL file is shown in figure 6.

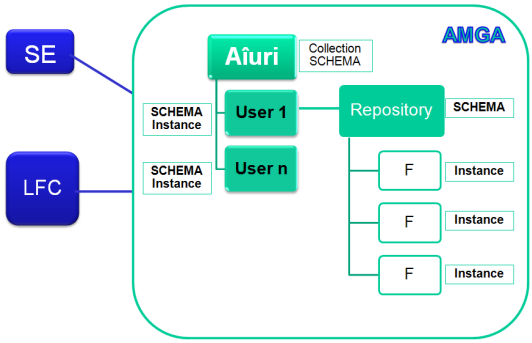

Figure 5a – Schematic example of the AMGA structure for Aîuri Project

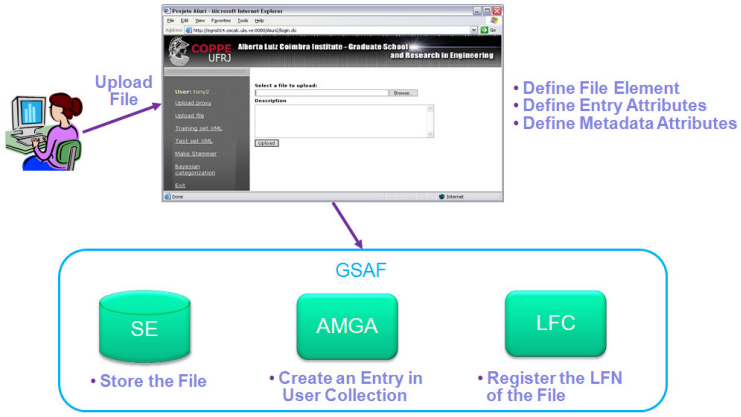

Figure 5b – Use of GSAF structure

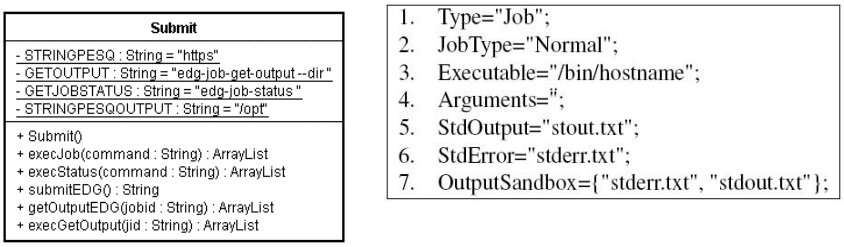

Figure 6 – Example of the GSAF architecture

It is necessary to remark that the work philosophy of the Aîuri Portal is to provide grid services to its users. In a preliminary approach, using the grid structure available at NACAD, all the grid services implemented are based on web services technology. In a second approach, using the EELA grid, the grid services are not based on web services, but its services are also enabled for the users by the portal.

6. Experiments

In order to evaluate the quality of the results obtained and, principally, to validate the environments, two experiments are presented below. Two sets of texts are used, taken from CETENFolha [15], which are previously classified by Computational Process of Portuguese Project. This previous classification is used to create a categorization model and later, to check the categorized results. The tests were carried with and without balancing. These tests were performed based on four distinct approaches: (i) stems and stop words were not used; (ii) stop words were used; (iii) stems and

stop words were both used; (iv) only stems were used. The graphics are used to illustrate the results of the models, with their corresponding f-measures and time processing for the best models created using a local computer and the NACAD Grid. The structure of the tests is shown in table 2.

Table 2. Structure of the experiments

Not Balanced			Balanced		
Topic	**Trainning**	**Test**	**Topic**	**Trainning**	**Test**
Brazil	2483	100	Brazil	1250	250
Money	2124	100	Money	1250	250
Sport	2594	100	Sport	1250	250
World	1565	100	World	1250	250

The experiments show that the best classification results were obtained with the class *Sport*. This is so, certainly, because the vocabulary in this class is quite distinct from the vocabulary of the remaining classes. During the experiments, it was observed that the use of stems, in average, caused an improvement of about 2% in the generated models. Both categorizers show similar results, as shown in figures 7a and 7b. However, the Bayesian categorizer, in average, performed faster, and obtained classification models slight better than those obtained by the linear score categorizer.

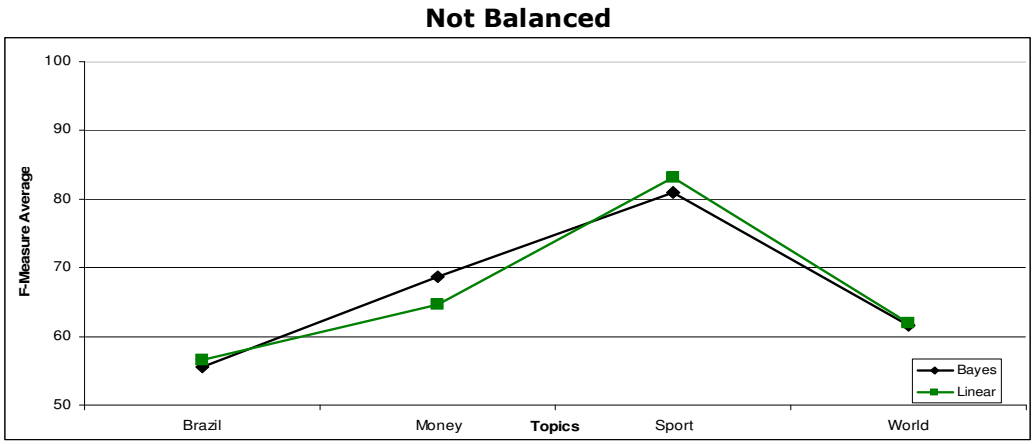

Figure 7a – Comparison of the results computed by the bayesian and linear score categorizers (not balanced)

Figure 7b – Comparison of the results computed by the bayesian and linear score categorizers (balanced)

7. Conclusion

We discuss the use of the Aîuri Portal for the execution of text mining grid services processed in grids environments.

The Aîuri Portal provides a very effective means of submission of algorithms to grid environments, since its functionalities encapsulate several tasks that, without the portal, would be performed by the user, making the use of computational grids accessible to ordinary users.

An additional advantage of our portal is its versatility, since two grid environments and the local machine are available to the user, and more environments can be incorporated to it.

The experiments show very good text mining processing times of grid submission compared to local submission, encouraging the continuity of the development.

The results obtained with the Bayesian algorithm were slightly better than those obtained with the linear score algorithm. The algorithms were tested using global dictionaries, which boosts the quality of the models when compared with the use of local dictionaries.

8. Acknowledgements

The authors would like to thank the High Performance Computing Center (NACAD) at the Graduate School and Research in Engineering (COPPE), Federal University of Rio de Janeiro (UFRJ) for providing the computational resources for this research.

The authors would like also to thank to EELA Project for supporting the gridification of text mining grid service in the context of EELA in EGRIS-2 (Second EELA Grid School).

The Aîuri Project is an application supported by the EELA-2 project, funded by the European Commission under the Grant Agreement #223797, where it would be developed and deployed..

9. References

[1] Lopes, M.C., Costa, M.C.A.; Ebecken, N.F.F., "Text Mining", In: Rezende, S.O. (Org.). Intelligent Systems: Foundations and Applications (in Portuguese). Editora Manole Ltda. 2002.

[2] Berman, F., Fox, G., Hey, T. "The Grid: past, present, future", Chapter 1, Berman, F., Fox, G., Hey, T., (eds) "Grid Computing: Making the Global Infrastructure a Reality", John Wiley & Sons, 2003

[3] Feldman, R. and Dagan, I., "Knowledge discovery in textual databases (KDT)", In Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95), Montreal Canada, AAAI Press, 1995

[4] Sebastiani, F., "A Tutorial on Automated Text Categorization", Proceedings of THAI-99, European Symposium on Telematics, Hypermedia and Artificial Intelligence, Italy, 1999

[5] Thomas, M. and Boisseau,J., "Building Grid computing portals: The NPACI Grid portal toolkit", Chapter 28 , in Berman, F., Fox, G., Hey, T., (eds) "Grid Computing: Making the Global Infrastructure a Reality", John Wiley & Sons, 2003

[6] Tzeras, K. And Hartman, S., "Automatic indexing based on bayesian inference networks". In Proceedings 16th ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93), pp. 22-34, 1993

[7] Lewis, D. D., Ringuette, M., "Comparison of two learning algorithms for text categorization". In Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94), 1994.

[8] Lacerda, W.S. ; Braga, A.P., "Experiments of a Pattern Classifier based on Naive Bayes Rule"(in Portuguese). INFOCOMP - Revista de Computação da UFLA, Lavras, v. 1, n. 3, p. 30-35, 2004.

[9] Qi, L., Jin, H., Foster, I., Gawor, J.: HAND: Highly Available Dynamic Deployment Infrastructure for Globus Toolkit 4. Document available at http://www.globus.org/alliance/publications/papers/HAND-Submitted.pdf

[10] Foster, I.: Globus Toolkit Version 4: Software for Service-Oriented Systems. In: Jin, H., Reed, D., Jiang, W. (eds.): IFIP International Conference on Network and Parallel Computing. Spring-Verlag, NPC 2005, LNCS 3779, 2-13.

[11] Weiss, S. M., Indurkhya, N., Zhang, T., Damerau, F. J., Text Mining: Predictive Methods for Analyzing Unstructured Information. New York, Springer Science+Business Media, 2005. 237p.

[12] Koblitz, B., Santos, N., "AMGA User's and Administrator's Manual, Nov 2006.

[13] Scifo, S., "GSAF-Grid Storage Access Framework"., Jun 2007.

[14] Pacini, F. Job Description Language – How To, Dec 2001.

[15]CentenFolha – Conjunto de Textos para Mineração de Textos URL http://acdc.linguateca.pt/cetenfolha/

[16] Serpa, A. A., "Aîuri: A Web Portal for Text Mining integrated to Computational Grids", M.Sc. Dissertation, COPPE/UFRJ, 2007