

# Grid Enabled problem solving environments for Text Categorization

Jian mei, Wu zhang and Suge wang  
Department of Computer Science and Technology, Shanghai University,  
Shanghai, 200072, China  
meijian\_2003@yahoo.com.cn

## Abstract

*As the large volume of resources involved and the power of computational Grids increased, there is a corresponding and urgent need for employ the grid technologies into problem solving environment (PSE) domain in order to improve the PSE to a better usability and higher performance. In this paper, we describe how to develop a framework of a Grid enabled PSE for text categorization with related Grid technologies. This grid-enabled PSE is able to support the activities that concern the building of the text classifier service, the classifying of the texts, the defining of the workflow, the selecting of service's Grid nodes and the reflection of the execution status through the web portal. Further, an example instance based on this Grid enabled system in the construction of PSE is shown in the end of this paper, and the performance of the PSE is evaluated in terms of the results of the application it generate.*

## 1. Introduction

The problem solving environment (PSE) is a system that provides all the computational facilities necessary to solve a target class of problems. It uses the language of the target class and users need not have specialized knowledge of the underlying hardware or software [1]. Examples of such systems include *AVS* [2], *Cactus* [3], etc. Owing to the rapid advancement of the Grid technologies, employing these technologies into building the PSE framework has become more and more significant. And as a result, we can add, delete or modify the PSE's components dynamically without changing the PSE framework. In the reference [4], they designed a **Grid PSE Builder** just using the Grid technologies for building the PSE.

In this paper we propose a novel Grid PSE framework for text categorization. Within this PSE platform, we can construct the text classifier and estimate the category of the new text. Our key contributions of this project include the following aspects: (1) we construct the PSE framework with the

Grid technologies, partition the process of text categorization and design the text categorization service. (2) According to the steps of the text categorization, we divide the system into two stages. The first stage is the building of the text classifier, and the second one is to employ the text classifier which is built on the first phase to classify the input texts. Furthermore, the feedback mechanism is employed on the second stage. (3) Just as other PSE, we also design a web portal on this Grid enabled PSE. Through the web portal, we can define and manage the workflow, select the service's Grid nodes and monitor the execution status of each service.

The rest of the paper is organized as follows. In Section 2, we simply describe the related work. We illustrate the Grid enabled PSE framework in detail in Section 3. In Section 4, we show an implementation of this PSE and Section 5 concludes the paper and discusses the future work.

## 2. The related work

In this section, we survey representative research on PSE for solving some problem belongs to different domain. A medical imaging PSE for advanced and Grid computing environments called *MedIGrid* is introduced in [5], which is a distributed application for the management, processing and visualization of biomedical images that integrates a set of software and hardware components, or, more specifically, a set of Grid collaborative applications useful to nuclear doctors.

For the purpose to calculate the partial differential equations (PDE) in numerical simulations research, a new distributed PSE, called *D-NCAS* [6], is proposed in order to support users to generate a computer program and to work on distributed computer systems. The PSE system inputs problem information including discretization and computation schemes, and outputs a program flow, a C-language source code for the problem and also a document for the program and for the problem. *PDE.Mart* is described in [7] which

solve a user-supplied PDE problem by using PDE solving services available on distributed Grid nodes.

So far, not many research on the text categorization with the grid enabled PSE are preceded yet. We design this framework in order that some novel research method on the text categorization can be merged into our architecture smoothly and seamlessly without changing the existed framework and interrupting the running workflows.

### 3. The framework of the Grid enabled PSE for text categorization

This PSE for text categorization contains mainly four components: Web Portal Service, Construct Text Classifier Service, Classify Text Proceeding Service and Workflow Management Service (see Fig.1).

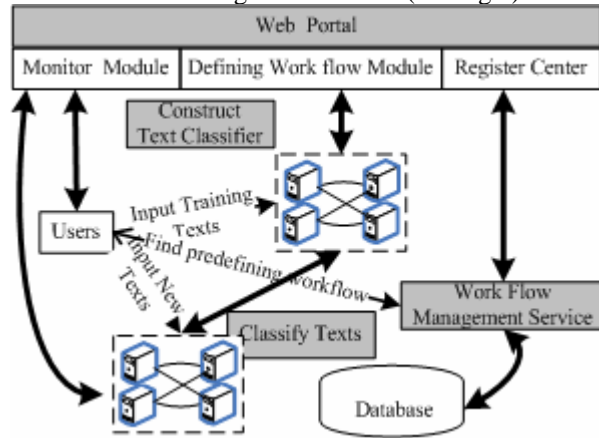


Fig. 1. The framework of the Grid enabled PSE for text categorization.

In our Grid enabled PSE, the text categorization process is focus on the two components, “Construct Text Classifier” and “Classify Text”. A whole process from the beginning of inputting new texts to the end of getting the classify results is divided into several independent modules, which are assigned to specific services offered by Grid nodes.

On solving a problem of text categorization, we deal with the texts on two stages. One is to construct the text classifier, and the other is to classify the texts. Firstly, the users log on the Web Portal, define a workflow and input the training texts into “Construct Text Classifier”. After the text classifier is constructed, the user input new texts into “Classify Text”. Finally, the “Classify Text” proceeds to classify the texts based on the text classifier built on the first stage and get the final classify results.

The Workflow Management Service is employed to record the information of the workflows, such as the service names, the related Grid nodes and the execution efficiency, etc. When the users define the workflow, they consider the Workflow Management Service as a reference.

#### 3.1. Web Portal Service

As Fig.1 is shown, this Web Portal Service includes Register Center, Monitor Module and Defining Workflow Module. Through the Register Center, the SC (Service Client) can register the own services into the PSE, become a SP (Service Provider) and provide the descriptions of the services to the Register Center. Each SP is belong to a Grid node, however, a Grid Node maybe provide different services. In this PSE once the SC would like to be a SP, it must download a middleware which is responsible for the execution status of workflow on this SP.

The Monitor Module is a real time module, which focuses are on presenting the states of all the SPs and the status of the running workflow. This module obtains the real time information through the middleware mentioned above and the execution status service which we will discuss in the following section. In this monitoring management, we make use of a web based monitoring tool (Map Center [8]) which provides access to status information.

Defining Workflow Module is aimed at define a workflow for a certain problem, the mainly work in this module is to choose the appropriate solution service, select the relevant Grid node and confirm the execution orders of the services. We define the workflow as follows:

$\langle \text{ServiceName}, \text{RunSN}, \text{GridNode}, \text{ExecutionOrder} \rangle$

**ServiceName** is the name of the service; **RunSN** is the concrete selected service which will be run; The **GridNode** means the name of Grid node providing this service; and the last **ExecutionOrder** is represented as the order of this service in the whole workflow of problem solving. Such as,  $\langle \text{Training Service}, \text{KNN Service}, \text{Grid\_MJ}, 8 \rangle$ . All the workflows can be stored in the Workflow Management Service.

#### 3.2. Construct Text Classifier Service

On constructing the text classifier, we segment the service into five sub services which are Text Representation Service, Feature Extraction Service, Calculation Features Weigh Service, Training Service and Threshold Value Selection Service (See Fig.2).

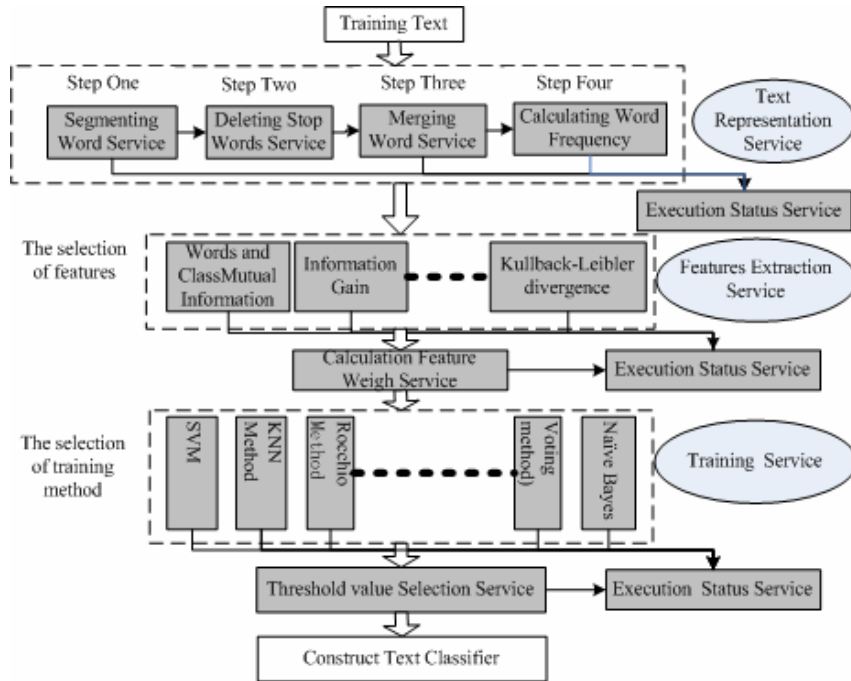


Fig. 2. The architecture and workflow of constructing the text classifier

We make many references to the text categorization system, and summarize the major task in the text representation stage. In Fig.2 four steps are list in the text representation service, and the four steps' execution is provided by the Grid nodes. However, in our Grid enabled PSE, we design one Grid node to provide all the four steps' service. The reason for doing so is to take the execution efficiency into consideration that the volume of the data between the four steps is very large.

As for the Feature Extraction Service, there exist some methods, such as Words and Class mutual information method, information gain method, Kullback-Leibler divergence, etc. Each method is designed to be a service and integrated within our PSE. We have regulated the input and output parameters in the "Register Center" module of the Web Portal Service. Users can select one method service from the Feature Extraction. The successive Calculation Feature Weigh Service is used to calculate the weigh of these features.

The mechanism of the Training Service is the same as the Feature Extraction Service. The user can select one training method service from the available Training Services. The Threshold Value Selection Service is employed to calculate the likelihood between the text and a certain category. When classifying a text, the text may belong to two or more than two categories. In this case, the service can

confirm the text's category with the category likelihood.

Each service's input information are the output results of the pre-service, and the own output results will be as the input information for the next service. After successive processes, finally the text classifier is constructed. All the information and the results are described by XML. If the workflow defined by the user is to be executed, an exclusive job ID is assigned. And when a service is to start or finish, the service's middleware will send a message to the Execution Status Service.

### 3.3. Classify Text Service

This service is based on the text classifier built in the service "Construct Text Classifier". During the course of the text classification, it will relate to four Grid services which are marked with the gray background as Fig.3 shown.

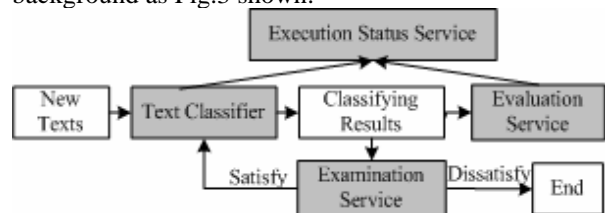


Fig. 3. The architecture and workflow of classifying text

Execution Status Service is the same as the one in the “Construct Text Classifier”, and provides the execution status of the workflow to the Web Portal. Evaluation Service serves with the performance of the workflow. We use the three traditional indexes of the text categorization, which are Precision, Recall and F1 [9]. Given a category labeled as  $i$ , assume that there are  $n_i$  test documents that belong to the category. Also, assume  $m_i$ , is the number of test documents classified to category  $i$  by our PSE, where  $l_i$  test documents are correctly classified. They are defined as the following formulas:

$$Precision_i = \frac{l_i}{m_i} \times 100\% \quad (1)$$

$$Recall_i = \frac{l_i}{n_i} \times 100\% \quad (2)$$

$$F1_i = \frac{Recall_i \times Precision_i \times 2}{Recall_i + Precision_i} \quad (3)$$

On the other hand, the feedback control mechanism is handled by the last service (Examination Service). The service is aimed at two aspects: one is to add the special and new words/characters which occur in the new text into the text classifier, and another aspect is to calculate the value of the new text categorization through some pre-defined roles, if the value is equal or more than the specify threshold value, the text category will be added into the text classifier or modify the existed categories of the text classifier. Now, we only

implement the first aspect which adds the new feature words/characters into the classifier. The second aspect is very complicated in establishing the examining principles, so we will do more research on this aspect in the future work

#### 4. A simple Implementation on this PSE

We test our Grid enabled PSE on a repository with 2930 Chinese texts. We select the Information Gain method as the “Feature Extraction Service” and KNN method as the “Training Service”. In the Fig.4, we describe the process of constructing text classifier and classifying text category.

In this text repository, all the texts are classified into 38 categories by the experts. In our experiment, we only make use of six categories’ texts which are about 500 texts. The key checkpoint of this experiment is to verify the consistencies between the results are drawn through our PSE and the ones are drawn from the existing text categorization system (*TxtCat* [10]). Here, we only list the results generated by our PSE as Table.1:

We do the same experiment based on the same training set and testing set on the *TxtCat* system. And the conclusion is the same as the above table. Through this experiment, we can verify the usability of this Grid enabled PSE framework, ensure the validity of the workflow of this PSE application and propose the future work.

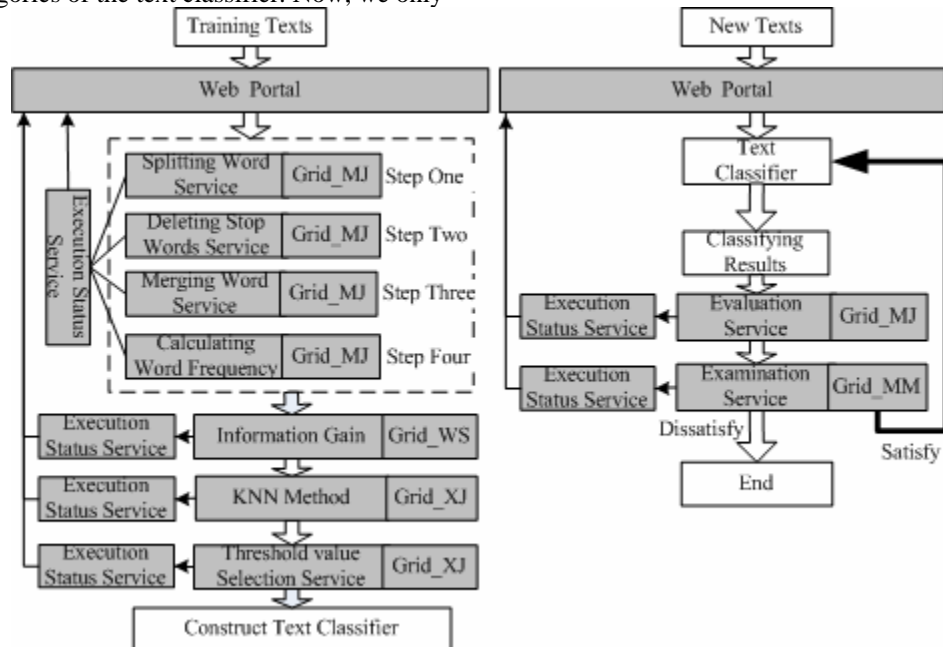


Fig. 4. An instance of Grid enabled PSE for text categorization. The left part describes how to build the text classifier based on the training set, and the right part is represented as the process of text categorization.

Table 1. The values of three index of text categorization

Name of Category	Precision	Recall	F1
Literature and Art	84.34%	84.56%	84.44%
Commerce and Economy	85.83%	86.67%	86.24%
Entertainment	83.45%	84.34%	83.89%
Government and Politics	87.08%	87.08%	87.08%
Society and Culture	82.39%	83.78%	83.08%
Education	89.11%	91.42%	90.25%

## 5. Conclusion and future work

This Grid enabled PSE facilitates the usability of the distributed computing resources, the various Feature Extraction method service and the diverse Training method service on the Grid. According to the characteristic of the text categorization, we divide the text categorization procedure into two stages: Construct Text Classifier and Classify Text. Also, this PSE framework integrates Test Representation services, Feature Extraction services, Training services, Calculation Weigh services, Execution Status service, Evaluation service and Feedback Control service. In this paper, we have described the workflow of the PSE in detail and introduce the functions of different sub-services briefly. We also present a simple instance to show the workflow and demonstrate the usability and extensibility of this PSE.

Additional future work of this project will be to integrate the existed and novel classifying method and training method. In addition, to extend the performance of data transfers, we intend to investigate protocols based on Quality of Service concerning the transformation of large quantities of data. Further, since on defining the workflow the optimal path selection is not considered in our PSE, we plan to add the path selection algorithm into consideration in the future research.

## References

[1] I.Foster and C.Kesselman (Eds): The Grid: Blueprint for a New Computing Infrastructure. [http://www.mkp.com/books\\_catalog/1-55860-475-8.asp](http://www.mkp.com/books_catalog/1-55860-475-8.asp), Morgan Kanfmann, Los Altos.  
 [2] Avs/advanced visual systems. <http://www.avs.com>.  
 [3] The cactus code server. <http://www.cactuscode.org>.  
 [4] Motonori Hirano, Naotaka Yamamoto and Hiroshi Takemiya: Grid PSE Builder: A Framework for Building Web-based Distributed PSE on Grid. HPCAsia, 2004.

[5] V.Boccia,M.R.Guarracino,L.D'Amore and G.Laccetti: A Grid Enabled PSE for Medical Imaging: Experiences on MedIGrid\*. 18th IEEE Symposium on Computer-Based Medical Systems (CBMS'05) 1063-7125  
 [6] Shigeo Kawata, Hideaki Fuju, and Hideaki Sugiura,etc. A Distributed Problem Solving Environment (PSE) for Scientific Computing.1st IEEE International Conference on e-Science and Grid Computing, Melbourne  
 [7] M. Mu. PDE.Mart: A network-based problem solving environment for PDEs. ACM Trans. Mathematical Software, 31(4), 2005.  
 [8] Map Center Home Page: <Http://mapcenter.in2p3.fr>  
 [9] Xinhao WANQ, Dingsheng LUO, Xihong WU and Huisheng CHI. Improving Chinese Text Categorization by Outlier Learning. Proceeding ofNLP-KE'05  
 [10] A text categorization system based on SVM and KNN. <http://www.nlp.org.cn>  
 [11] You Li-ping, Wang Su-ge. Rules and distributions of Chinese verb-verb collocations[J]. Computer Engineering and Applications. 2005,41(23):179-181 (in Chinese with English abstract).  
 [12] Wang Su-ge You Li-ping, , Liu Kai-ying. Automatic acquisitive method of verb-verb collocations[C]. in Maosong Sun, Tianshun Yao, Chunfa Yuan, eds., Advances in Computation of Oriental Languages, Proceedings of 20th International Conference on Computer Processing of Oriental Languages, Tsinghua University Press, August, 2003  
 [13] Li Rong-luL,Wang Jian-hui,Chen Xiao-yun,Tao Xiao-peng, Hu Yun-fa. Using maximum entropy model for Chinese text categorization[J]. Journal of Computer Research and Development 2005,42(1):94-101. (in Chinese with English abstract).  
 [14] I.Foster and C.Kesselman (Eds): The Grid: Blueprint for a New Computing Infrastructure. [http://www.mkp.com/books\\_catalog/1-55860-475-8.asp](http://www.mkp.com/books_catalog/1-55860-475-8.asp), Morgan Kanfmann, Los Altos, CA,1988.  
 [15] Ann Chervenak, Ian Foster, Carl Kesselman, Charles Salisbury and Steven Tuecke: The Data Grid: Towards an Architecture for the distributed Management and Analysis of Large Scientific Datasets. Journal of Network and Computer Applications, 23:187-200, 2001.  
 [16] G. von Laszewski, I. Foster, et al, Designing Grid-based Problem Solving Environments and portals, 34th Hawaii International Conference on System Science, 2001.