A Novel Granularity-based Classification in Cloud Environment

Wenjuan Shao [1, 2,a], Qingguo Shen [2,b], Xianli Jin [3,c] and Liaoruo Huang[2,d]

[1] Zijin College, Nanjing University of Science and Technology, Nanjing, Jiangsu 210023 - China

[2] College of Communications Engineering, PLA University of Science and Technology, Nanjing, Jiangsu 210007 - China

[3] Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu 210003 – China

[a] shaowj_nj@139.com, [b] shenqg2016@163.com, [c] jxl@njupt.edu.cn, [d] huangliaoruo@sina.com

Keywords: Classification; Cloud Computing; Granularity; Non-uniform granularity

Abstract. Cloud computing service is a new computing paradigm which consists of distributed and large scale computing resources. Effective classification managements for the resources is necessary. In this paper, we describe some concept and principle about classification, and present a classification algorithm based on non-uniform granularity. Experiments results carried on the blog posts illustrate the effectiveness of the new algorithm.

Introduction

Cloud computing is a set which consists of massive amount and various types of computing resource, virtual machine monitor, and data. In cloud environment, we need storage cloud to save the data. Moreover, more and more enterprise applications such as e-mail, World Wide Web and some other on-line services usually require high-performance online data querying and processing.

A number of machine-learning techniques have been used in the classification for the resource s[1-5]., such as Support Vector Machine (SVM), Naïve-Bayes (NB), and k-nearest neighbor (KNN) classifier. Although information granularity approach is rarely researched on resources classification, it has been applied into many other fields, such as rough set theory[6] divide and conquer[7],, machine learning, cluster analysis, databases etc[8]. We extend the work above by proving that granularity principle can be used to resolve resources classification problems[9].

In this paper, we introduce a novel granularity-based classification in cloud environment. Experiments results on the blog posts verify the effectiveness of the algorithm.

Granularity principle for classification

Clustering is a multivariate and statistical method which can classify sample points. We use the shortest distance method as similarity measure function to generate clustering pedigree chart, the specific process is summarized as follows, we take the sample points in Fig.1 as an example:

(1) Regarding every sample point as one class.

(2) Calculating the distance between the sample points.

(3) Merging the closest pair to be a new class, regarding the distance of the merged classes as the height of the new class and recalculating the distance between the new class and other classes .

(4) According to the name and the distance of merged sample points, marking them on corresponding location in clustering pedigree chart.

(5) Returning (3), until all sample points can merge into one class.

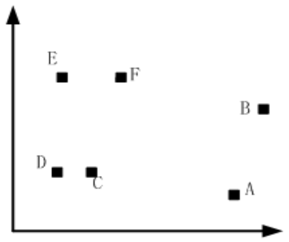

Fig. 1 Sample points

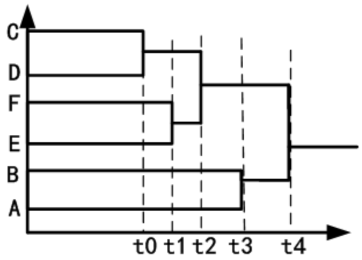

Fig. 2 Clustering pedigree chart for sample points

We assume classification threshold named *t*, and the given object set *X*={*A*,*B*,*C*,*D*,*E*,*F*}.The clustering pedigree chart for sample points in Fig.1 is showed in Fig.2 .We can see that when *t* varies, the corresponding classification results will differ greatly, for example:

(1) If *t*>=t4, all sample points can be clustered into one class;

(2) If t1<*t*<t2, then object set *X* are clustered into four classes {*A*}, {*B*}, {*E*, *F*}, and {*C*, *D*};

(3) If *t*<t0, every sample point can form a class respectively, {*A*}, {*B*}, {*C*}, {*D*}, {*E*}, {*F*}.

Specifically, with a lager *t* , these sample points will show us a "rough" outline ,for some details are ignored, slightly analogous points will also be clustered into the some class; while with a smaller *t*, some minor differences between sample points are portrayed clearly, and only extremely analogous sample points can be clustered into the same class.

Classification based on non-uniform granularity in cloud environment

1. Framework of the algorithm

Firstly, we do clustering algorithm to the cloud resources, clustering pedigree chart will be obtained, by using classification threshold to cut it, we get different branches. Then, we repeat cutting with finer threshold (granularity) until stopping conditions are satisfied. Finally, we can get classification results.The framework is shown as Fig. 3.

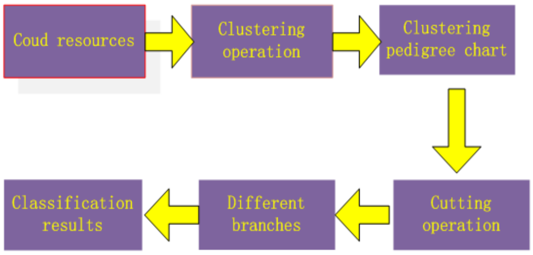

Fig. 3 Framework for non-uniform granularity classification

2. Algorithm description

According to the features of cloud resources, we can get a granular system *G*=(*U*, *D*, *F*), where *G* denotes resources pool, *U* defines all resources in the resources pool, in this paper it specifies video , audio, text and blog posts etc., *D* is the description set of the resources, and F defines the relationship between the resources.

Algorithm *CNUGCE*: A classification based on non-uniform granularity in cloud environment

Inputs: A classification task which refers to the resource classification in storage cloud, classification threshold *T*.

Outputs: Classification results of the resources.

Step 1. perform clustering algorithm to the resources to get clustering pedigree chart。

Step 2. use threshold *T* to cut the clustering pedigree chart, some branches are obtained.

Step 3. repeat step 2 , until the leaf nodes of each branch only belong to one class.

The size of threshold *T* represents granularity, firstly we choose coarse granularity as much as possible when cutting. Nevertheless, as for the elements in the boundary, since their difference is not easy to be seen in large-grained world, and that they are more likely to cause confusion with elements of other classes ,so a relatively much finer granularity is needed, which can distinguish them clearly.

The results of *CNUGCE* on Fig. 2 are shown as Fig. 4.

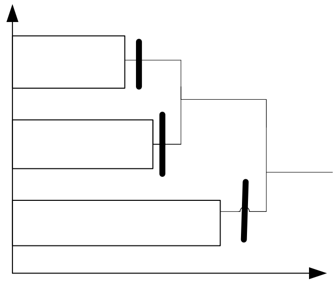

Fig. 4 Classification results based on non-uniform granularity

From Fig. 4, we find that when classification threshold is larger than a certain value, sample instances of red, blue, and green color are all classified into one class. Whereas with slight small *T*, some minor differences between sample instances can be well portrayed. The smaller threshold *T* can classify the sample instances into three classes correctly, namely green class, blue class and red class.

Experiment

Our dataset have 2472 blog posts in total. And we use category tree structure of Wikipedia, which mainly consists of 12 categories, as is shown in Table 1. We select training set and test set according literature[10], blog corpus above will be divided into 12 parts, one of which is selected as an open test set, and the remaining 11 parts are defined as training set and closed test set, make sure each of them can be an open test set turn by turn. A total of 12 times classification operations will be performed to calculate the average accuracy for classification. Experimental results are summarized in Table 1.

Table 1. Experimental results

Category	Category size	Open Test Accuracy	Close Test Accuracy
Arts and Culture	247	78.40%	93.8%
Physics and Nature	100	82.50%	92.4%
Philosophy and Thinking	64	87%	90.2%
Geography and Geology	120	82.10%	92.5%
History and Events	60	83.50%	91.7%
Mathematics and Logics	26	90.10%	92.6%
Society and Sciences	530	89.70%	94.0%
Technology and Sciences	528	82.80%	96.2%
Health and Fitness	92	83.40%	90.5%
Military	25	78.90%	90.9%
Sports	150	88.50%	93.1%
Economics	530	80.30%	95.7%

In order to validate the performance of the proposed algorithm, we compared the *CGUGCE* with NB, KNN, SVM. The results are shown as Table 2 .

Table 2. Comparing the *CNUGCE* between several algorithms

Category	Category size	SVM	NB	KNN	GNUGCE
Arts and Culture	247	91.5%	92.4%	93.5%	93.8%
Physics and Nature	100	90.6%	90.5%	91.8%	92.4%
Philosophy and Thinking	64	88.6%	86.6%	85.5%	90.2%
Geography and Geology	120	92.3%	90.8%	91.8%	92.5%
History and Events	60	88.9%	89.6%	90.5%	91.7%
Mathematics and Logics	26	86.7%	85.5%	87.5%	92.6%
Society and Sciences	530	93.7%	91.9%	93.3%	94.0%
Technology and Sciences	528	95.6%	94.5%	95.5%	96.2%
Health and Fitness	92	89.6%	88.3%	89.5%	90.5%
Military	25	85.6%	86.8%	89.5%	90.9%
Sports	150	92.8%	91.3%	92.5%	93.1%
Economics	530	95.6%	93.4%	94.8%	95.7%

Seen from the results of experiments, we can get the following conclusions:

(1) Seen from Table 1, classification algorithm for blog posts is effective. In fact, especially for the closed tests, the accuracy of *CNUGCE* can be more than 90%.

(2) Seen from Table 2, we find that the proposed algorithm can improve classification performance for the blog posts efficiently.

(3) Moreover, classification performance is sensitive to the size of granularity (classification threshold *T*). When granularity is gradually increasing, the corresponding category varieties of *CNUGCE* will decrease gradually. And with the decreasing of granularity, classification will become fine, and the category varieties will increase. But when granularity is smaller than a certain value, it will lead to insignificance of the classification result, blog posts which have large similarity measure will be wrongly classified into different categories.

Conclusion and Future Work

For the problem of massive amount and various types of resources in cloud environment, we present an efficient classification algorithm based on non-uniform granularity. Clustering algorithm is used to generate clustering pedigree chart. And most resources can be classified into correct class by modifying classification threshold *T* (granularity) to cut the clustering pedigree chart. The size of *T* is vital to the performance of the proposed algorithm. Through comparing with traditional algorithms, we find that our proposed algorithm can improve the performance of classification. Future work will be considered on using the classification results above to search literature.

Acknowledgement

This work was financially supported by the Natural Science Foundation of China (Project No. 61271254).

References

[1] Rey-Long Liu. Interactive high-quality text classification [J].

[2] T Joachims, "Text categorization with Support Vector Machines: Learning with many relevant features," Machine Learning: ECML-98, vol. 1398/1998, pp. 137-142, 1998.

[3] H.T. Ng, W.B. Goh, and K.L. Low, “Feature selection, perceptron learning, and a usability case study for text categorization,” Proc. of the Int. ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 67-73, 1997.

[4] A McCallum and K Nigam, "A Comparison of Event Models for Naïve Bayes Text Classification," AAAI-98 Workshop on "Learning for Text Categorization", 1998.

[5] N Friedman, D Geiger, and M Goldszmidt, Bayesian Network Classifiers, Machine Learning, vol. 29, 1997, pp. 131-163.

[6] L.A. Zadeh, Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, fizzySets and System, 19, 111-127, 1997.

[7] G. Y. Wang, Rough set theory and knowledge acquisition, Xi’an:Press of Xi’an Jiaotong University, 2001.

[8] Shao Jian. Information granularity computing based on rough sets[MS dissertation]. Institute of Automatics, Chinese Academy of Sciences, Beijing, 2000.

[9] L.A. Zadeh, fizzy sets and information granularity, in: Advances in fuzzy Set Theory and Applications, Gupta, N., Ra, gde, R. and Yager, R. (Eds.), North-Holland, Amsterdam, pp. 3-18, 1979.

[10] HuangXuan-Jing. Retrieval, classification and summarization of large scale Chinese text [PhD dissertation]. Fudan University, Shanghai 1998(in Chinese).