A Novel Approach for Automatic Web Page Classification using Feature Intervals

J. Alamelu Mangai, Dipti D. Kothari and V. Santhosh Kumar

**Department of Computer Science, BITS Pilani, Dubai Campus,
DIAC, Dubai 345055, U.A.E**

Abstract

A new web page classification algorithm using weighted voting of feature intervals known as WVFI is proposed in this paper. This classifier first discretizes the web page features using a supervised disctretization algorithm which identifies the number of intervals each feature has to be discretized automatically. Each feature is then made to predict the class of the corresponding feature in the test web page using the class distribution of its intervals. The final class of the test web page is predicted by aggregating the weighted vote of each feature. Experiments done on a benchmarking data set called WebKB has shown good classification accuracy when compared with many of the existing classifiers.

Keywords: *Web Page Classification, Discretization, WebKB, weighted voting.*

1. Introduction

Web page classification, WPC, also known as web page categorization is a process of classifying web pages into meaningful category labels. The applications of WPC are as follows:

- Web directories provided by different search engines like Google, Yahoo etc can be constructed, maintained or expanded using advanced WPC techniques.
- WPC can improve the quality of search results. When a user types in a particular keyword, the numbers of relevant results are increased through WPC.
- A question answer system uses WPC techniques to improve the quality of answers.
- Web content filtering is another application of WPC.

Web pages can be classified by two methods: syntactic and semantic. This article emphasizes syntactic classification, which uses a set of words or patterns in a web page to classify it. The alternative approach uses natural language analysis of the text. The techniques to pre process natural language and extract the text semantics are too costly for large amounts of data. In addition, they are only effective with well structured text, a thesaurus and other conceptual information. Web pages are represented using vector-space model which is commonly used in information retrieval. The web page data after preprocessing in the vector space model are continuous in nature depending on the weights of each feature in the web page collection. Machine learning algorithms have been applied in real-world classification tasks like WPC. Many of these algorithms focus on learning in discrete feature space. In the case of continuous attributes, there is a need for a discretization algorithm that transforms continuous attributes into discrete ones, or to use a different algorithm. Also, algorithms which can handle both continuous and discrete features perform better with the discrete-valued attributes. Discrete values play an important role in data mining and knowledge discovery. Many studies have shown that induction tasks can benefit from discretization: rules with discrete values are normally shorter and easy to understand and discretization can lead to improved predictive accuracy [1]. Apart from the algorithmic requirements, discretization also helps in increasing the speed and accuracy of induction algorithms. It makes the results of the induced classifier shorter, compact and easier to understand than those generated using continuous features. This paper proposes a method to classify web pages by discretizing the web page features. The output predictions are made by combining the weighted vote of each feature interval. The rest of the paper is organized as follows: Section 2 highlights the related work, proposed work is described in Section 3, and details of the experiments done are summarized in Section 4. Section 5 highlights the results and findings.

2. Related work

Many approaches for automatic web page classification have been witnessed over years in literature. The performance of the web page classifiers are improved from different perspectives, namely by dimensionality reduction (feature selection), using the word occurrence statistics in a web page (content based), using the

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

relationship between different web pages (link based), using the association between queries and web pages (query log based) and by using the structure of the page, the images, links contained in the page and their placement (structure based).

Various methods are adopted in literature to select the best features for WPC. CfsSubset evaluator is combined with term frequency [2] to obtain a minimum number of highly qualitative features. Three distinct features of a web page namely, the URL, title and meta data which are believed to have more predictive information about a web page are used [3] with machine learning methods to classify a web page. The output of PCA principal component analysis is combined with a manual weighting scheme to classify web pages using neural networks [4]. A fuzzy ranking analysis with discriminating power measure [5], rough set theory [6] and an integrated use of ant colony optimization with fuzzy-rough sets [7] is used to reduce the dimensionality of web pages. A new feature selection method that incorporates hierarchical information about the categories is used by the authors in [8]. This prevents the classifying process from going through every node in the hierarchy. On – page features (content based) and features of neighboring pages (context based) are also used [9] to classify a web page.

By exploiting the characteristics of Chinese web pages, a new feature selection method by assigning weights to the HTML tags is proposed in [10]. The structure of the web pages is used to classify them into information, research and personal home pages [11]. Blocks in [12] are units that compose a web page namely, paragraphs, tables, lists and headings. The association between these blocks, web pages and the queries are used to frame a query with content based classification framework to classify a web page. Visual features of a web page like color and edge histograms, Gabor and texture features [13] summaries generated by human experts are used in [14]. These approaches of web page classification cannot be applied in situations which suffer from hardware and software limitations. Further, they require lot of human expertise and are computationally complex. The various technologies that can be explored in web information extraction have been explored in [15] and the authors have expressed their concern that many researchers start with the complex approaches directly rather than trying out the simpler ones first.

To summarize, clustering approaches are computationally expensive, meta tags cannot be used, since there is a possibility for the web page author to intentionally include keywords which do not reflect its content, merely to increase its hit-rate. Link based and structure based approaches also fail in scenarios to correctly classify a web page from its print version, since there is no link in it and not all web pages contain images. Motivated by these facts this paper proposes a method for WPC using the content of the web page. Designing a web page classifier involves substantially more amount of preprocessing, since web pages have additional challenges over pure text documents. It is stated in [16] that classification accuracy can be improved with discretization on data sets including continuous features. Classification by voting feature intervals, VFI is implemented in [17]. Intervals are constructed around each class for each attribute (basically discretization). Class counts are recorded for each interval on each attribute. Classification is by voting. A simple attribute weighting scheme is added to VFI in [18]. Higher weight is assigned to more confident intervals, where confidence is a function of entropy. The vote of each feature interval is combined to make output predictions. As an attempt to improve the performance of the machine learning web page classifiers, in this paper we have implemented a new algorithm for WPC using the discrete values of the web page features. The proposed method in this paper first discretizes features using a supervised discretized algorithm mentioned in our earlier work [19]. It then combines the weight of a feature with the vote made by the feature using its interval to predict the class of the test web page. Many discretization techniques exist in literature, like Simple Binning, Chi-Square , etc. A recent survey of them is presented in [20] .Some of them requires the user to specify the number of intervals each feature needs to be discretized. Methods like Chi-Square are based on a significance threshold as a stopping criterion. But no definite rule is given to choose this threshold. But the discretization method used in this paper identifies the number of bins and the stopping criterion automatically. Intuitively, it may occur that discretization reduces the accuracy of the discovered knowledge. It may cause some relevant detailed information to be lost. However, a discretization algorithm can make global decisions, based on all the values of the attribute to be discretized. This is in contrast with the local handling of continuous attribute values typically found in rule-induction algorithms.

3. Proposed work

3.1 Architecture of WPC Framework

The architecture of the proposed method of Web Page Classification (WPC) framework is as shown in Fig. 1.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

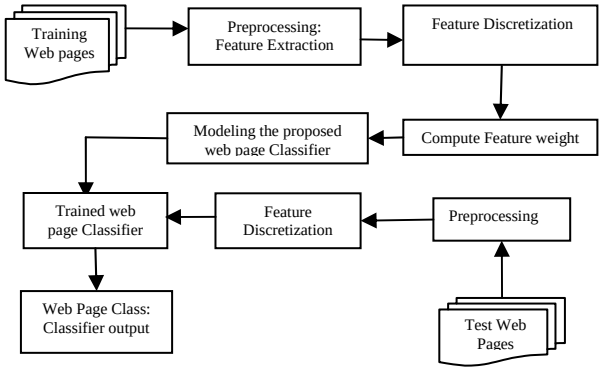

Fig. 1 Architecture of the proposed Web Page Classifier

Classification is a three step process namely, 1) Discretize the web page features 2) Compute the weight of each feature and 3) Train the classifier

3.2 Discretize the web page features

As web page feature values are continuous in nature, they are first discretized using a discretization algorithm that is presented in our earlier work [19]. The discretization process is highlighted below.

Input: Web page Feature Vectors (***WFV***), Web page Classes(***C***), the threshold B_{size}, I_{min} the inconsistency threshold within an interval.

Output: Discretized web page Feature Vectors, ***DWFV***

Algorithm:

A.1 for each web page feature ***f*** in ***WFV*** do

a. Sort the values of ***f*** into ascending order with their corresponding class labels.

b. If two consecutive values in ***f*** belong to two different classes, establish a cut point between them. Let C_1 be the set of all such cut points.

c. for each bin of values ***B*** in C_1 do

c.1 Find the majority class of the bin, B_{maj}.

c.2 Merge two consecutive bins, if their corresponding size is less than B_{size} i.e., $|B_{maj}| < B_{size}$

c.3 Save the new cut points in C_2.

d. for each bin in C_2 do

d.1 Find the inconsistency measure B_I of two consecutive bins $B_I = \frac{|B| - |B_{maj}|}{|B|}$,where, $|B_{maj}|$ is the number of feature values in the bin that belong to the majority class and $|B|$ is the number of feature values in a bin.

d.2. Merge two consecutive bins with the same majority class, and also have inconsistency measure within I_{min}

d.3 Save the new cut points in C_3

d.4 Assign a label to each cut point in C_3

d.5 Replace the web page feature values in ***f*** with the corresponding bin label.

e. Write the discretized web page feature vectors with their corresponding class labels in ***DWFV***

f. Stop.

3.3 Compute Feature Weights

Each feature is assigned a weight which is its information gain [18] with respect to the class. The feature which is more predictive of the class has more information gain value. This value helps in assigning more weight to the vote given by the most predictive feature.

3.4 Train the Proposed Web page Classifier

C.1 for each feature, ***f*** in ***DWFV*** do

a. Find the number of intervals, ***n*** of ***f*** and the respective label of each interval

$f_{value} = \{f_{value1}, f_{value2}, \ldots . f_{valuen}\}$

b. for each label in f_{value} do

b.1 find the class distribution of this label

b.2 find the majority class of this label, f_{value_maj}, which is the vote of this feature interval,

3.5 Testing Phase of the Proposed Web Page Classifier

D.1 for each feature, f_{test} in test web page do

a. Find the majority class of f_{test} using the vote given by the corresponding feature interval from training phase.

D.2 Combine the vote of each feature interval with the weight of the respective feature to get the predicted class of the test web page, $C^{**} = \sum_{j=1}^{m} weight_j \; x \; vote_j$, where ***m*** is the number of features in the web page.

4. Experiments and Results

Experiments were done on a bench marking data set called WebKB [21]. This data set contains WWW-pages collected from computer science departments of various universities. The pages are manually classified into the following categories student, faculty, staff, department, course, project and others. For the analysis of the proposed work course web pages are considered as positive examples and student web pages as negative examples The HTML tags, stop words, punctuations, digits and hyphens are removed from the web pages in our preprocessing phase. Then, the web pages are subsequently stemmed. The final set of features is extracted using tf-idf weighting as stated in our earlier work [22]. The results of feature extraction are shown in Table 1 where Input Size stands for the number of positive and negative examples of web pages used in our experiments.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

Table 1: Results of feature Extraction

Input Size	No. of Instances	No. of Features
50 – 50	38	5
70- 30	56	5
100 – 100	92	6
200 – 200	291	13
300 – 200	298	9
300 – 300	414	14
350 – 150	391	13
400 – 200	422	15
400 – 300	557	17
400 – 400	585	17

After feature extraction the web pages are stored in arff, attribute relation file format which is supported by WEKA [18], a machine learning tool. Figure 2 shows a portion of the 70-30 input file after feature extraction.

Viewer

Relation: webkb-CFS-DT-1-Rnd-DR-r6

No.	assign Numeric	cours Numeric	cse Numeric	document Numeric	ithaca Numeric	class Nominal
1	0.0	0.0	0.0	0.0	0.73	student
2	0.4	0.2	0.73	0.73	0.0	course
3	0.4	0.17	0.73	0.0	0.0	course
4	0.0	0.0	0.81	0.0	0.0	course
5	0.0	0.0	0.75	0.0	0.0	course
6	0.4	0.0	0.0	0.0	0.0	course
7	0.5	0.21	0.79	0.57	0.0	course
8	0.0	0.2	0.0	0.0	0.0	course
9	0.4	0.22	0.0	0.57	0.0	course
10	0.5	0.17	0.7	0.0	0.0	course
11	0.52	0.23	0.0	0.0	0.0	course
12	0.4	0.2	0.75	0.57	0.0	course
13	0.4	0.2	0.81	0.67	0.0	course
14	0.53	0.2	0.81	0.57	0.0	course
15	0.54	0.21	0.0	0.0	0.0	course
16	0.0	0.17	0.0	0.0	0.0	course
17	0.4	0.17	0.82	0.64	0.0	course
18	0.0	0.17	0.0	0.0	0.81	student
19	0.55	0.19	0.0	0.0	0.0	course
20	0.45	0.19	0.0	0.0	0.0	course
21	0.0	0.21	0.0	0.0	0.0	course
22	0.0	0.0	0.77	0.64	0.0	course
23	0.0	0.0	0.0	0.0	0.81	student
24	0.0	0.17	0.0	0.0	0.73	student
25	0.47	0.23	0.0	0.57	0.0	course

Fig. 2 The 70 – 30 input file after feature extraction

As the first phase of the proposed work, the web pages are discretized as explained in section 3.1. The results of discretization are then stored in arff format. Figure 3 shows the portion of the 70 – 30 file after discretization. Each feature is discretized into a certain number of bins which is automatically identified by the proposed method. As seen in Fig 3 each feature is discretized into a certain number of intervals.

Viewer

Relation: webkb-CFS-DT-1-Rnd-DR-NumericClass-r6

No.	ASSIGN Nominal	COURS Nominal	CSE Nominal	DOCUMENT Nominal	ITHACA Nominal	class Nominal
1	3	7	11	15	20	student
2	6	10	14	18	19	course
3	6	7	14	15	19	course
4	3	7	14	15	19	course
5	3	7	14	15	19	course
6	6	7	11	15	19	course
7	6	10	14	18	19	course
8	3	10	11	15	19	course
9	6	10	11	18	19	course
10	6	7	14	15	19	course
11	6	10	11	15	19	course
12	6	10	14	18	19	course
13	6	10	14	18	19	course
14	6	10	14	18	19	course
15	6	10	11	15	19	course
16	3	7	11	15	19	course
17	6	7	14	18	19	course
18	3	7	11	15	20	student
19	6	9	11	15	19	course
20	6	9	11	15	19	course
21	3	10	11	15	19	course
22	3	7	14	18	19	course
23	3	7	11	15	20	student
24	3	7	11	15	20	student
25	6	10	11	18	19	course

Fig. 3 The 70 – 30 input file after the first phase of the proposed web page classifier.

As a second step of the proposed method, the weight of each feature is calculated from its information gain. Table 2 shows the weight of each feature in the 70 – 30 input file.

Table 2: The weight of each feature in the 70-30 input file

Feature Index	Feature's Rank	Feature	Feature Weight
5	1	ITHACA	0.5436
3	2	CSE	0.1042
1	3	ASSIGN	0.1029
2	4	COURS	0.0362
4	5	DOCUMENT	0.0306

It can be seen from the file, the feature with the highest information gain means, it is more predictive of the category of the web page. And so, it has the highest rank. In our proposed web page classifier, instead of assigning equal significance to the vote of all features, the vote given by features with highest information gain are weighted more. The proposed classifier is implemented both with and without weighting and the results of the same are shown in Table 3. The classifier is modeled using percentage split method with 70% of the input data as training and the remaining 30% for testing the classifier.

Table 3: . Accuracy of the Proposed Web Page Classifier with and without feature weighting

Input Size	Voting without weighting	Weighted Voting
50 – 50	100	100
70 – 30	88	100

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

100 – 100	85	85
200 – 200	57	93
300 – 200	78	88
300 – 300	57	91
350 – 150	76	90
400 – 300	67	96
400 – 400	67	94

The results in Table 3 show that there is a significant improvement in classification accuracy with weighted voting than with no weighting of feature votes. The proposed web page classifier is then compared with many other existing classifiers, namely, rule-based classifiers (oneR and decision table, DT), decision-tree based classifier (J48), Naïve Bayes classifier (NB), instance based classifier (kstar), support vector machine based classifier (SVM), ensemble classifier (boosting), combined decision table and Naïve Bayes (DTNB) and voting based on feature intervals (VFI). Previous studies have shown that rule-based, decision tree based and NB are well known for their simplicity. Instance based classifiers have proven to work well in classifying textual documents. The SVM and boosting classification algorithms have the ability to handle large scale data.

Table 4 shows the classification accuracy of the various classifiers. As seen from Table 4, based on average classification accuracy, the proposed web page classifier performs better than VFI and oneR. Its performance is also equally as good as the other classifiers. Fig 4 shows the classifier accuracy graphically.

Table 4: Accuracy of the various Web page Classifiers

Input Size	Proposed classifier	One R	DT	J48	NB	Kstar	SVM	Boosting	DTNB	VFI
50 – 50	100	100	100	100	100	100	100	100	100	100
70 – 30	100	100	100	100	100	94.11	100	100	100	82.35
100-100	85	92.85	92.85	92.85	92.85	89.28	92.85	92.85	92.85	82.14
200–200	93	78.16	87.35	87.35	90.80	97.70	95.40	87.35	93.10	82.75
300–200	88	87.64	94.38	96.62	95.50	95.50	94.38	93.25	96.62	74.15
300–300	91	81.45	91.12	90.32	94.35	98.38	96.77	92.74	93.54	94.35
350–150	90	87.17	94.87	94.87	95.72	97.43	96.58	94.87	94.87	85.47
400–300	96	79.64	89.22	94.01	94.01	95.80	97	91.61	96.40	80.83
400–400	94	78.28	92.57	92	96	96.57	93.71	94.28	97.14	83.28
Average	**93**	**87.24**	**93.59**	**94.22**	**95.47**	**96.08**	**96.29**	**94.10**	**96.05**	**85.03**

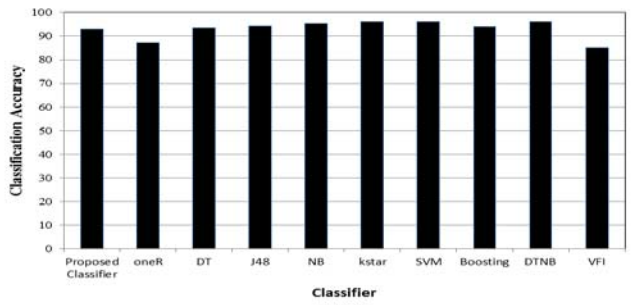

Fig. 4 Accuracy of the various Web page Classifiers.

However, it has the following advantages as compared to other existing classifiers.

- It is simple and easy to interpret.
- It is an eager learner. It need not compare the test data with every training data as in nearest neighbor classification algorithm.
- As features are discretized, this helps in reducing the resources utilized by the classifier during the modeling phase. Only the feature intervals and their respective majority class have to be stored.
- Makes global decision based on all values of the discretized feature, as opposed to local decision based on the numeric value of a feature as in rule-induction algorithms.

5. Conclusions

Due to the exponential increase in the volume of the World Wide Web, WWW, automatic web page classification is indeed immensely required to assist the search engines. Content based web page classification has shown promising results in this direction. In this paper a new web page classifier which combines the predictions made by all features is proposed. Each feature votes for a particular class using the class distribution of its intervals. From the experimental results it is inferred that the performance of this classifier is good when compared with other existing web page classifiers. Also, it is easy to understand and implement. Our future work is to run this classifier for multiple categories of web pages.

References

[1] H. Liu, F. Hussain, C. L. Tan and M. Dash, "Discretization: An enabling technique", Data Mining and Knowledge Discovery, Vol. 6, No. 4, 2002, pp. 393-423.

[2] M. Indra Devi, R. Rajaram R, K. Selvakuberan, "Generating the best features for web page classification", Webology, Vol 5, No. 1, 2008.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

[3] L. W. Han and S. M. Alhashmi, "Joint Web-feature (JFEAT): A Novel Web page Classification Framework", Communications of the IBIMA, Vol. (2010), Article ID 73408, 2010.

[4] A. Salamat, S. Omata, "Web page feature selection and Classification using neural networks", Information Science ACM, Vol. 158, No. 1, 2004, pp. 69-88.

[5] C-M Chen, H-M Lee, Y-J Chang, "Two novel feature selection approaches for web page classification", Expert systems with Applications, Vol. 36, 2009, pp.260-272.

[6] O. Wakaki, H. Itakura, M. Tamura, "Rough Set-Aided Feature Selection for web page classification", in IEEE/WIC/ACM International Conference on Web Intelligence, 2004, pp.70-76.

[7] R. Jensen and Q. Shen, "Web page classification with ACO-Enhanced Fuzzy-Rough Feature Selection", LNCS, Vol. 4259, 2006, pp.147-156.

[8] X. Peng, Z. Ming and H. Wang, "Text learning and Hierarchical Feature Selection in Web page Classification", LNCS, Advanced Data Mining and Applications, Vol. 5139, 2008, pp. 452 – 459.

[9] M. Farhoodi, A. Yari, M. Mahmoudi, "A Persian Web Page Classifier Applying a Combination of Content-Based and Context-Based Features", International Journal of Information Studies, Vol. 1, No. 4, 2009, pp. 263 – 271.

[10] W. Chen, Y. Du, P. Zhang, B. Han, "The Effective Classification of the Chinese Web pages based on kNN", JCIS, Vol. 6, 2010, pp. 2925-2932.

[11] A. P. Asirvatham, K. K. Ravi, "Web Page Classification based on Document Structure", Awarded Second Prize in National Level Student Paper Contest conducted by IEEE India Council, 2001.

[12] W. Dai et al., "A Novel Web Page Categorization Algorithm Based on Block Propagation Using Query-Log Information", Advances in Web-Age Information Management Lecture Notes in Computer Science, Vol. 4016, 2006, pp. 435-446.

[13] V. de Boer, M. V. Someran, "Classifying Web Pages with Visual Features", WEBIST (1), 2010, pp. 245-252

[14] D. Shen, Z. Chen, Q. Yang, H-J Zeng, B. Zhang, Y. Lu, W-Y Ma, "Web-page classification through summarization", in Proceedings of the 27th annual international ACM SIGIR conference on Research and Development in Information Retrieval, 2004, pp. 242-249.

[15] D. Xhemali, C. J. Hinde, and R. G. Stone, "Embarking on a Web Information Extraction Project", in Proceedings of UK Conference on Computational Intelligence, 2007.

[16] M. Hacibeyoglu, A. Arslan, S. Kahramanli, "Improving Classification Accuracy with Discretization on Data Sets Including Continuous Valued Features", WASET, 2011.

[17] G. Demiröz and H. A. Güvenir, "Classification by voting Feature Intervals, Machine Learning", LNCS, Vol. 1224, 1997, pp. 85-92

[18] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update". SIGKDD Explorations, Vol. 11, No. 1, 2009.

[19] J. A. Mangai, D. D. Kothari and V. S. Kumar, "A Supervised Discretization Algorithm for Web Page Classification", in International Conference on Innovations in IT, 2012, pp. 226 - 231

[20] S. Kotsiantis, D. Kanellopoulos, "Discretization Techniques: A recent Survey", GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 47 – 58.

[21] The 4 Universities data set [Online]. Available: http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/, Accessed June 2012

[22] J. A. Mangai, V. S. Kumar, "A Novel Approach for Web Page Classification using Optimum features", IJCSNS, Vol 11, No.5, 2011, pp. 252 – 257.

J. Alamelu Mangai graduated from Annamalai University, India in 2005. She is currently pursuing her Ph.D. from BITS Pilani, Dubai Campus. She is currently working as a Senior Lecturer in the Department of Computer Science in BITS Pilani, Dubai Campus, U.A.E, since 2006. Her research interests include data mining algorithms, text and web mining.

Dipti D. Kothari is a final year student of computer science at BITS Pilani, Dubai Campus. She graduates in August 2012. Her research interests include mathematics, data mining and computer networks.

V. Santhosh Kumar received his Ph.D. degree from Indian Institute of Science, Bangalore, India. He is currently working as Assistant Professor at BITS Pilani, Dubai Campus, U.A.E. His research interests include Data Mining and Performance Evaluation of Computer Systems.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.