Evaluating the Ambiguities of Class Structure via Euclidean Distance

Jing-Doo Wang
Department of Computer Science and Information Engineering
Asia University
jdwang@asia.edu.tw

Abstract

The classification is a supervised learning approach in the machine learning and, therefore, the class structure was specified by the domain experts manually in advance. The goal of this paper is to evaluate the degree of ambiguity between any two classes in the existing class structure while the similarity between two classes was estimated via Euclidean distance. In this paper, *Distinguishable Distance Ratios (DDR)* and *Class Ambiguity Ratio (CAR)* between any two classes are proposed to indicate the degree of the ambiguity between classes. The degree of class ambiguity between two classes supposed to be high if the value of DDR is low and the value of CAR is high. The experimental resources for class structure evaluation includes "Iris Plant", "Wine Recognition" and "Glass Identification", and the DDR and CAR did reveal the degree of class ambiguity. This works offer domain expertise an approach to examine the fitness of class structure if necessary.
Keywords: Classification, class structure, class ambiguity.

1 Introduction

The classification is a supervised learning approach in the machine learning[1, 10, 19]. There are a lot of approaches applied to the classification problem, such as Linear classifier, Decision Tree, Neural Network, K-Nearest- Neighbor, and Support Vector Machine(SVM)[1] and so forth. All of above approaches aimed to improve the performance of classification according to accuracy, recall, precision, F_1 measure, etc. However, the class structure was specified by the domain experts manually in advance. As the amount of instances in the classification increased, it might result in reorganizing the class structure to form new classes because of the characteristics of these instances within the same class being too diverse. That situation usually happened in the portal sites. On the other hand, the class label of a new instance might be suitable to many classes when it came; meanwhile, the curator has to assign that instance to only one of classes. As the time passed, there existed many instances belonging to multi-classes[14, 15]. These cases mentioned above lead to the problems that how to evaluate the fitness of class structure and when to reorganize the existing class skeleton .

There are studies[4, 6, 11, 13, 12, 20, 21] for structure discovery or taxonomy construction. In [11], Punera et al. proposed a framework, called "CLUMP", for unsupervised discovery of structure in data, and explored the problem of learning n-ary tree based hierarchies of categories with no user-defined parameters[12, 13]. Chuang and Chien [4] tackled the problem of taxonomy generation for diverse text segments using the Web as an additional knowledge source. Gao et al.[6] mined for a hierarchical structure from the flat taxonomy of a data corpus by consistent bipartite spectral graph copartitioning. Zhang et al.[21] developed a tool, named "InfoAnalyzer", to assist the enterprise to prepare large set of samples used for text classification, and proposed an automatic method of collecting training samples to build hierarchical taxonomies[20]. In [4], Chuang and Chien generated the taxonomy for diverse text segments using the Web as an additional knowledge source. Gates et al.[7] proposed a system for generating a large general-purpose taxonomies and categorization system. Wang et al.[18] propose the idea of class proximity and cast the hierarchical classification as a flat classification with the class proximity modeling the closeness of classes. Above studies, however, seldom mentioned or discussed the problem whether the original class structure was suitable for all instances or not. It is worthy of having a function to evaluate the fitness of class structure just like the criterion functions for document clustering[22].

The goal of this paper is to evaluate the degree of ambiguity between any two classes in the existing class structure in the vector space model, and the similarity

between two vectors was estimated via Euclidean distance. In this paper, *Distinguishable Distance Ratios (DDR)* and *Class Ambiguity Ratio (CAR)* between any two classes are proposed to indicate the degree of the ambiguity between classes. The degree of class ambiguity between two classes supposed to be high if the value of DDR is low and the value of CAR is high. To verify whether the values of DDR and CAR could interpret the existence of class ambiguity or not, the confusion matrix H[19] was constructed using SVM classifier and each resources were divided into two thirds for training and the other for testing. It might exist ambiguous regions between two classes if there are many instances in one class were miss-classified to another class with the classifier that is esteemed to achieve high classification accuracy, such as the SVM classifier [1, 19]. The program "easy.py" in the LIBSVM[3] was adopted to tune the parameters of SVM to achieve the best classification accuracy in this paper.

The experimental resources includes "Iris Plant", "Wine Recognition" and "Glass Identification" [2]. There are 3 types of "Iris plant" as, "Setosa", "Versicolour" and "Virginica". Experimental results showed that the class boundary between "Versicolour" and "Virginica" might be ambiguous. The "Wine Recognition" was a well posed problem with "well behaved" class structures in a classification context[2], and the statements above coincided with the experimental results that each value of DDR was high, that of CAR was low, and the accuracy achieved by SVM was as high as 96.61%. The study of "Glass Identification" was motivated by criminological investigation[2], and there were 7 classes in the "Glass Identification". The first 4 classes were "window glass" and the other were "non-window glass". The former, furthermore, could be divided into "float-processed and "non-float-processed". Experimental results showed that the closeness of the relationship between these 7 classes as described above could be found.

The remainder of this paper is organized as follows. Section 2 gives the notations and the computation of DDR and CAR. Section 3 gives experimental results. Section 4 gives conclusions and discussions.

2 Methods

Distance(similarity) functions play a centrol role in all classification/clustering algorithm. How to evaluate the distance (similarity) of two instance vectors is one of the fundamental steps of classification in the vector space model[16]. The most commonly used distance functions for numeric attributes are Euclidean distance, Manhattan(ciy block) distance, Minkowski distance. For text documents, a document is usually considered as a "bag" of words while the position information of words are ignored. Thus the similarity is used to compare two documents rather than the distance, and the most commonly used similarity function is the cosine similarity[19]. In this paper the Euclidean distance was adopted and defined as follow. Let $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$ be two n-dimensional vector. The Euclidean distance $dist(X, Y)$ between X and Y is defined as $\sqrt{\sum_{d=1}^{d=n}(x_d - y_d)^2}$.

To evaluate the ambiguity between classes, first of all, the values of *Distinguishable Distance Ratios (DDR)* and *Class Ambiguity Ratio (CAR)* between all pairs of any two classes are computed. Then the confusion matrix constructed by SVM classifier was used to verify whether these two values, DDR and CAR, could tell the ambiguity or not. The confusion matrix $H = (h_{i,j})$ is obtained via SVM classifier by partition each resources into two thirds for training and the other for testing. Each entry h_{ij} in H indicates the number of instances in C_i that were miss-classified to C_j if $i \neq j$. That is, the value of h_{ij} is expected to be high when the value of DDR is low and the value of CAR is high. The definitions of some notations are given in the section 2.1. The computation processes of DDR and CAR are given in the section 2.2 and 2.3, respectively.

2.1 Notations

Let C be the set of predefined classes, and $|c|$ be the number of predefined classes. Let C_i be the set of instances in the training set that belong to the ith class, and F_j be the set of instances in the testing set which are classified to the jth class. $|C_i|$ and $|F_j|$ are the number of instances in C_i and F_j, respectively. Let $H_{i,j}$ be the set of instances in C_i that is classified to F_j. That is, $H_{i,j} = C_i \cap F_j$. Let $h_{i,j} = |H_{i,j}|$. Note that $C_i = \bigcup_{j=1}^{j=|c|} H_{i,j}$ and $F_j = \bigcup_{i=1}^{i=|c|} H_{i,j}$. The confusion matrix $H = (h_{i,j})$, as shown in Table 1, consists of the statistics of the classified documents.

The set I consists of instances from $C_1, C_2, \ldots, C_{|c|}$, and C_i, $1 \leq i \leq |c|$, consists of class-labeled instances, $I_{C_i,1}, I_{C_i,2}, \ldots, I_{C_i,|C_i|}$, where $|C_i|$ is the number of instances in C_i. Each instance is represented as one n-dimensional vector. For example, $I_{C_i,q}$, the qth instance in C_i, is represented as $(t_{(C_i,q),1}, t_{(C_i,q),2}, \ldots, t_{(C_i,q),n})$. The M_{C_i} is the centroid of all instance vectors in C_i and is defined as

Table 1: The confusion matrix H.

	C_1	C_2	$\cdots$	C_j	$\cdots$	$C_{\|C\|}$
C_1	$H_{1,1}$	$H_{1,2}$	$\cdots$	$H_{1,j}$	$\cdots$	$H_{1,\|c\|}$
C_2	$H_{2,1}$	$H_{2,2}$	$\cdots$	$H_{2,j}$	$\cdots$	$H_{2,\|c\|}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$		$\vdots$
C_i	$H_{i,1}$	$H_{i,2}$	$\cdots$	$H_{i,j}$	$\cdots$	$H_{i,\|c\|}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$		$\vdots$
$C_{\|C\|}$	$H_{\|c\|,1}$	$H_{\|c\|,2}$	$\cdots$	$H_{\|c\|,j}$	$\cdots$	$H_{\|c\|,\|c\|}$

$\frac{\sum_{q=1}^{q=|C_i|} I_{C_i,q}}{|C_i|}$.

The $dist(X, M_{C_i})$ is the Euclidean distance from the instance X to the centroid M_{C_i}. Let $M_{D(C_i)}$ and $S_{D(C_i)}$ be the mean and the standard deviation of the distances from every instances in C_i to the centroid M_{C_i}, and are given as $M_{D(C_i)} = \frac{\sum_{X \in C_i} dist(X, M_{C_i})}{|C_i|}$ and $S_{D(C_i)} = \sqrt{\frac{\sum_{X \in C_i} (dist(X, M_{C_i}) - M_{D(C_i)})^2}{|C_i|}}$, respectively. Let $N_{i,j}(k) = \{X | dist(X, M_{C_i}) < k * S_{D(C_i)}, X \in C_j, k \geq 1\}$ be the set of instances X in C_j whose values of $dist(X, M_{C_i})$ are smaller than k times the values of $S_{D(C_i)}$. Let $|N_{i,j}(k)|$ be the number of these instances and $P_{i,j}(k) = \frac{|N_{i,j}(k)| - |N_{i,j}(k-1)|}{|C_j|}$ be the probability of instances X in class C_j whose values of $dist(X, M_{C_i})$ range from k to $(k-1)$ times values of $S_{D(C_i)}$.

2.2 Distinguishable Distance Ratios(DDR)

Let *Distinguishable Distance Ratio (DDR)* between C_i and C_j be as following:

$$DDR(C_i, C_j) = \frac{CCD(C_i, C_j)}{\gamma * (S_{D(C_i)} + S_{D(C_j)})}. \quad (1)$$

where $CCD(C_i, C_j) = dist(M_{C_i}, M_{C_j})$, *Class Centroid Distance (CCD)*, is the Euclidean distance from M_{C_i} to M_{C_j} and γ is a given constant. In this paper the value of γ was set to be 3 because instances that lie three or more standard deviations from the centroid are considered to be outliers, assume the distribution of the values of each instance vector to its centroid is a normal distribution[8].

Intuitively, the two classes would be separable if two cetroids are far from one another and the distribution of instances vectors in its class are dense. That is, it is easier to distinguish from two classes if their centroids are far from one another. The larger value of the $DDR(C_i, C_j)$ is, the more distinct boundary between C_i and C_j is. On the other hand, it might be

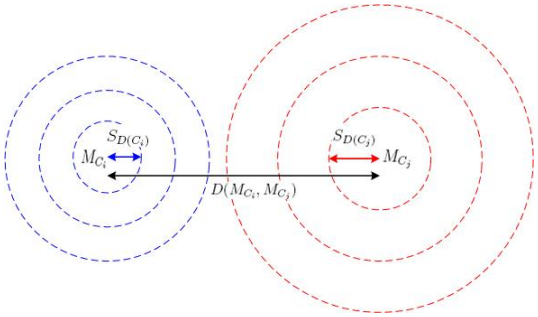

Figure 1: $DDR(C_i, C_j) > 1$ when $\gamma = 3$

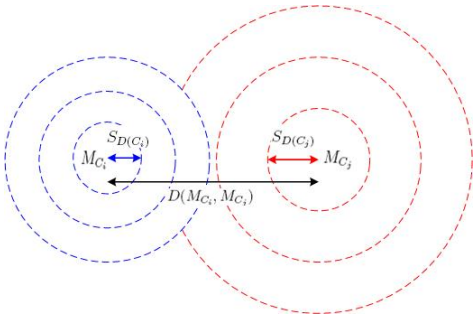

Figure 2: $DDR(C_i, C_j) < 1$ when $\gamma = 3$

hard to identify the boundary between two classes if the distribution of the instances vectors in these two classes are loose. The distribution of the instances vectors in C_j related to C_i is measured by inspecting the distribution of these distances $dist(X, M_{C_i})$, for all $X \in C_j$, based on the scale of the $S_{D(C_i)}$. That is, the larger value of $(S_{D(C_i)} + S_{D(C_j)})$ is, the less distinguishable boundary between C_i and C_j is. In this paper, as shown in the figure 1, the boundary of two classes, C_i and C_j, might be distinguishable, otherwise, it might be ambiguous as shown in the figure 2. Note that the relation of DDR between C_i and C_j is symmetric, that is, $DDR(C_i, C_j) = DDR(C_j, C_i)$.

2.3 Class Ambiguity Ratio (CAR)

The *Class Ambiguity Ratio (CAR)* of C_j relative to C_i is given as following:

$$CAR(C_i, C_j) = \frac{CA(C_i, C_j)}{CA(C_i, C_i)}. \quad (2)$$

where $CA(C_i, C_j) = \sum_{k=1} P_{i,j}(k) * \frac{1}{k^2}$ is the *Class Ambiguity (CA)* of class C_j relative to class C_i. The function of $\frac{1}{k^2}$ is to evaluate the weighting of the instances falling within the range of distances from k to $(k-1)$ times values of $S_{D(C_i)}$ such that it decreases

Table 2: The statistics of the experimental resources

Data Set	# of classes (|cl|)	# of instances (|I|)		# of features (n)
		Training	Testing	
Iris Plant	3	100	50	4
Wine Recognition	3	119	59	13
Glass Identification	7	143	71	9

Table 3: The parameters for LIBSVM achieve best accuracy for each resource

Data Set	Accuracy	C	g
Iris Plant	92%(=46/50)	2	8
Wine Recognition	96.61%(=57/59)	2048	0.000488281
Glass Identification	71.83%(=51/71)	33554432	0.000488281

as the distance to the centroid M_{C_i} increases.

To normalize the value of CAR of C_j relative to C_i, the value of $CA(C_i, C_i)$ is used as the denominator such that the values of the CARs are less than 1 because the probability of instances in C_i falling close to M_{C_i} is usually larger than that of instances in the other classes. Intuitively, the larger probability of the instances of C_j falling close to the centroid M_{C_i}, the more ambiguous boundary of C_j related to C_i is. Note that the relation of CA between C_i and C_j might not be symmetric. That is, $CA(C_i, C_j) \neq CA(C_j, C_i)$.

3 Experimental Results

The experimental resources includes "Iris Plant", "Wine" and "Glass Identification" [2] as shown in the Table 2. Each instances is represented as an vector in which the format of each attribute of that vector is the same as in [3]. In this study, the cells were marked as "yellow" in the table 4, 5 and 7 if their corresponding values of h_{ij} is greater than 1 in this study. To achieve the best classification accuracy using SVM classifier, the program "easy.py" in LIBSVM[3] was used for tuning parameters, C and g, for the best classification accuracy. The best accuracy and the corresponding parameters achieved for each resource were listed in the Table 3. We discussed the class ambiguities according to each resource as following.

3.1 Iris Plant

The "Iris Plant" was the most popular data sets for machine learning since 2007. There are 3 types of "Iris" as, "Setosa", "Versicolour" and "Virginica". As described in [2], one class is linearly separable from the other 2; the latter are not linearly separable from each other. As shown in Table 4, the values of DDR and

Table 4: The statistics of Iris Plant

		Setosa	Versicolour	Virginica
DDR	Setosa	0.00	1.26	1.97
	Versicolour	1.26	0.00	0.64
	Virginica	1.97	0.64	0.00
		Setosa	Versicolour	Virginica
CAR	Setosa	1.00	0.02	0.01
	Versicolour	0.17	1.00	0.60
	Virginica	0.04	0.18	1.00
		Setosa	Versicolour	Virginica
Confusion Matrix	Setosa	15	0	1
	Versicolour	0	15	2
	Virginica	0	1	16

Table 5: The statistics of Wine

		Class 1	Class 2	Class 3
DDR	Class 1	0.00	1.00	1.68
	Class 2	1.00	0.00	1.02
	Class 3	1.68	1.02	0.00
		Class 1	Class 2	Class 3
CAR	Class 1	1.00	0.25	0.16
	Class 2	0.28	1.00	0.31
	Class 3	0.11	0.20	1.00
		Class 1	Class 2	Class 3
Confusion Matrix	Class 1	19	0	0
	Class 2	1	22	1
	Class 3	0	0	16

CAR between two classes, "Versicolour" and "Virginica", were 0.64 and 0.6, respectively. It showed that the class boundary between "Versicolour" and "Virginica" might be ambiguous. This observation coincided with that there were 2 instances of "Versicolour" miss-classified to "Virginica".

3.2 Wine Recognition

As mentioned in [2], these data for "Wine Recognition" are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. It was said that the "Wine Recognition" was a well posed problem with "well behaved" class structures in a classification context. The above statements coincided with the experimental results, as shown in the Table 5, that the value of DDR is high(all above 1), the value of CAR is low (under 0.31), and the best accuracy achieved by SVM is as high as 96.61%(= 57/59).

Table 6: The description of class attribute of "Glass Identification"

	class attribute
c1	building_windows_float_processed
c2	building_windows_non_float_processed
c3	vehicle_windows_float_processed
c4	vehicle_windows_non_float_processed (none in this database)
c5	containers
c6	tableware
c7	headlamps

3.3 Glass Identification

The study of classification of types of glass was motivated by criminological investigation[2] because, at the scene of the crime, the glass left can be used as evidence if it is correctly identified. The description of class attribute of glass is given in the Table 6, and the first 4 classes, $c1$, $c2$, $c3$ and $c4$ were "window glass", the other were "Non-window glass". Note that $c4$ did not exist in this resource. As shown in the Table 7, among $c1$, $c2$ and $c3$, the values of DDR are low and the values of CAR are high as comparing the values of DDR and CAR to that among the other 3 classes, $c5$, $c6$ and $c7$. It means that $c1$, $c2$ and $c3$ are highly correlated with each other and this observation agrees with the description in the Table 6 that $c1$, $c2$ and $c3$ are the same type of glass as "window".

Among the group of "window glass", on the other hand, both of $c1$ and $c3$ were the types of windows with float-processed, while $c2$ was that without float-processed. It implied that $c1$ and $c3$ were closer to each other than they to $c2$. The above statement could be concluded from the experimental results that, when comparing to the values of the other classes in the row, the values of $DDR(c1, c3)$(the same as $DDR(c3, c1)$) was 0.09 and was the smallest value; the values of $CAR(c1, c3)$ and $CAR(c3, c1)$ were 1.09 and 0.87 which were the largest value in the corresponding rows, respectively.

Table 7: The statistics of Glass Identification

		c1	c2	c3	c5	c6	c7
DDR	c1	0.00	0.12	0.09	0.63	0.90	0.86
	c2	0.12	0.00	0.12	0.40	0.46	0.55
	c3	0.09	0.12	0.00	0.59	0.77	0.78
	c5	0.63	0.40	0.59	0.00	0.39	0.34
	c6	0.90	0.46	0.77	0.39	0.00	0.47
	c7	0.86	0.55	0.78	0.34	0.47	0.00
		c1	c2	c3	c5	c6	c7
CAR	c1	1.00	0.73	1.09	0.09	0.24	0.09
	c2	1.14	1.00	1.21	0.16	0.28	0.12
	c3	0.87	0.67	1.00	0.08	0.22	0.10
	c5	0.34	0.39	0.35	1.00	0.72	0.56
	c6	0.28	0.29	0.28	0.31	1.00	0.30
	c7	0.12	0.13	0.13	0.18	0.32	1.00
		c1	c2	c3	c5	c6	c7
Confusion Matrix	c1	15	6	2	0	0	0
	c2	2	22	0	0	1	0
	c3	3	2	1	0	0	0
	c5	0	1	0	3	0	0
	c6	0	0	0	1	2	0
	c7	1	0	0	0	1	8

4 Conclusions and Dicussions

This works offer domain expertise an approach to examine the fitness of class structure if necessary. In this paper, the values of *Distinguishable Distance Ratios (DDR)* and *Class Ambiguity Ratio (CAR)* were proposed to evaluate the ambiguity between classes in the vector space model via Euclidean distance. To verify whether these two values, DDR and CAR, could tell the ambiguity or not, the confusion matrix $H = (h_{i,j})$ constructed by SVM classifier was computed. Each entry h_{ij} in H indicates the number of instances in C_i that were miss-classified to C_j if $i \neq j$. That is, the value of h_{ij} is expected to be high when the value of DDR is low and the value of CAR is high in this study. The experimental results showed that the DDR and CAR did indicate the ambiguity between classes when comparing to the contents of the confusion matrix achieved by SVM classifier. To achieve the best classification accuracy using SVM classifier, the tool "easy.py" in LIBSVM is used for tuning parameters, C and g, for the best classification accuracy. Note that the SVM is one of well-known classifiers achieving high classification accuracy.

There might be a problem using Euclidean distance in the vector space model when the dimension n is very large in some applications, e.g. text classificatoin[5, 17], that the distances between two instance vectors were almost the same. However, this problem might be solved using the dimension reduction by *Principle Component Analysis (PCA)* or *Singular Value Decomposition (SVD)*[1]. On the other hand, the centroid representation works well if the classes are of the hyper-shperical shape. However, centroids may not be suitable if classes are elongated or are of other shapes[9]. It is our future work to experiment with the resource vectors in higher dimension and the other representation of class to make our con-

clusions more robust.

Acknowledgments. This work was partially supported by the research project of Asia University under grant $97-I-07$. Thanks for Prof. Hsiang-Chuan Liu and Prof. Jiunn-I Shieh for discussions about the statistics. Thanks for reviewer's valuable comments.

References

[1] Ethem Alpaydin. *Introduction to Machine Learning*. The MIT Press, 2004.

[2] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[3] Chih-Chung Chang and Chih-Jen Lin. Libsvm : a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm., 2001.

[4] Shui-Lung Chuang and Lee-Feng Chien. Taxonomy generation for text segments: A practical web-based approach. *ACM Trans. Inf. Syst.*, 23(4):363–396, 2005.

[5] Ronen Feldman and James Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, December 2006.

[6] Bin Gao, Guang Feng, Tao Qin, and Qian-Sheng Cheng. Hierarchical taxonomy preparation for text categorization using consistent bipartite spectral graph copartitioning. *IEEE Trans. on Knowl. and Data Eng.*, 17(9):1263–1273, 2005.

[7] Stephen C. Gates, Wilfried Teiken, and Keh-Shin F. Cheng. Taxonomies by the numbers: building high-performance taxonomies. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 568–577. ACM, 2005.

[8] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques, 2nd ed.* Morgan Kaufmann, 2005.

[9] Bing Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)*. Springer, January 2007.

[10] Tom M. Mitchell. *Machine Learning*. The McGraw-Hill Companies, Inc, 1997.

[11] Kunal Punera and Joydeep Ghosh. CLUMP: A scalable and robust framework for structure discovery. In *ICDM '05: Proceedings of the Fifth IEEE International Conference on Data Mining*, pages 757–760. IEEE Computer Society, 2005.

[12] Kunal Punera, Suju Rajan, and Joydeep Ghosh. Automatically learning document taxonomies for hierarchical classification. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 1010–1011. ACM, 2005.

[13] Kunal Punera, Suju Rajan, and Joydeep Ghosh. Automatic construction of n-ary tree based taxonomies. In *ICDMW '06: Proceedings of the Sixth IEEE International Conference on Data Mining - Workshops*, pages 75–79. IEEE Computer Society, 2006.

[14] Cédric Raguenaud, Martin Graham, and Jessie Kennedy. Two approaches to representing multiple overlapping classifications: A comparison. In *SSDBM '01: Proceedings of the Thirteenth International Conference on Scientific and Statistical Database Management*, page 239. IEEE Computer Society, 2001.

[15] Cédric Raguenaud and Jessie B. Kennedy. Multiple overlapping classifications: Issues and solutions. In *SSDBM '02: Proceedings of the 14th International Conference on Scientific and Statistical Database Management*, page 77. IEEE Computer Society, 2002.

[16] Baeza-Yates Ricardo and Ribeiro-Neto Berthier. *Modern Information Retrieval*. Addison Wesley, 1999.

[17] Jing-Doo Wang. *Design and Evaluation of Approaches for Automatic Chinese Text Categorization*. PhD thesis, Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan 62107, R.O.C., 2002.

[18] Ke Wang, Senquiang Zhou, and Shiang Chen Liew. Building hierarchical classifiers using class proximity. In *Proceedings of VLDB-99, 25th International Conference on Very Large Data Bases*, pages 363–374. Morgan Kaufmann Publishers, 1999.

[19] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*. Elsevier, 2005.

[20] Li Zhang, Tao Li, Shixia Liu, and Yue Pan. An integrated system for building enterprise taxonomies. *Inf. Retr.*, 10(4-5):365–391, 2007.

[21] Li Zhang, ShiXia Liu, Yue Pan, and LiPing Yang. Infoanalyzer: a computer-aided tool for building enterprise taxonomies. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 477–483. ACM, 2004.

[22] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis, 2001.