A Supervised Clustering Method for Text Classification

Umarani Pappuswamy, Dumisizwe Bhembe,
Pamela W. Jordan, and Kurt VanLehn

Learning Research and Development Center,
3939 0'Hara Street, University of Pittsburgh,
Pittsburgh, PA 15260, USA
umarani@pitt.edu

Abstract. This paper describes a supervised three-tier clustering method for classifying students' essays of qualitative physics in the Why2-Atlas tutoring system. Our main purpose of categorizing text in our tutoring system is to map the students' essay statements into principles and misconceptions of physics. A simple `bag-of-words' representation using a naïve-bayes algorithm to categorize text was unsatisfactory for our purposes of analyses as it exhibited many misclassifications because of the relatedness of the concepts themselves and its inability to handle misconceptions. Hence, we investigate the performance of the k-nearest neighborhood algorithm coupled with clusters of physics concepts on classifying students' essays. We use a three-tier tagging schemata (cluster, sub-cluster and class) for each document and found that this kind of supervised hierarchical clustering leads to a better understanding of the student's essay.

1 Introduction

Text Categorization (or Classification)[1] can be seen either as an Information Retrieval task or a Machine Learning task of automatically assigning one or more well-defined categories or classes to a set of documents. Starting with the work of Maron [1] in the early 60s, Text Classification (TC) has found a significant place in a variety of applications including: automatic indexing, document filtering, word sense disambiguation, and information extraction. Our main focus is on the machine learning aspect of TC with the goal to devise a learning algorithm capable of generating a classifier which can categorize text documents into a number of predefined concepts. This issue has been considered in several learning approaches both with a supervised learning scheme [2, 3] and with an unsupervised and semi-supervised learning scheme [4, 5, 6].

In its simplest form, the text classification problem can be formulated as follows: We are given a set of documents $D = \{d_1, d_2, d_3 \ldots d_n\}$ to be classified and $C = \{c_1, c_2, c_3, \ldots c_n\}$ a predefined set of classes and the values $\{0, 1\}$ interpreted as a decision to file a document d_j under c_i where 0 means that d_j is not relevant to the class defined and 1 means that d_j is relevant to the class defined. The main objective here is to devise a

[1] We prefer the term `Text Classification' to `Text Categorization' and hence use the same in the rest of our paper.

A. Gelbukh (Ed.): CICLing 2005, LNCS 3406, pp. 704 – 714, 2005.

learning algorithm that will be able to accurately classify unseen documents from D (given the training set with the desired annotations in the case of supervised learning).

In our paper, we describe a three-tier clustering method for classifying students' essay strings in the Why2-Atlas system. The students' essays are the answers to qualitative questions of physics. The task of the classifier is to map these essay strings into the corresponding principles and misconceptions of physics. A simple `Bag-Of-Words (BOW)' approach using a naïve-bayes algorithm to categorize text was unsatisfactory for our purposes of analyses as it exhibited many misclassifications because of the relatedness of the concepts themselves and its inability to handle misconceptions. Hence, we investigate the performance of k-nearest neighborhood algorithm coupled with pre-defined clusters of physics concepts on classifying students' essays. Though there have been many studies on word clustering for language modeling and word co-occurrence [7], very little work has been done on word/concept clustering for document classification.

We present the results of an empirical study conducted on a corpus of students' essay strings. The approach uses a three-tier tagging schemata (cluster, sub-cluster and class) for each document. Let C and SC refer to the Cluster and Sub-cluster respectively, and `Class (Cl)' refers to the actual principle or misconception being identified. Thus, C in the original definition now takes the form: $C = \{(C_1, SC_1, Cl_1), (C_n, SC_n, Cl_n)\}$. This kind of supervised clustering approach helps us to reduce the dimensionality of the texts and thereby leads to a better understanding of the student's essay.

The next section, namely Section 2 describes text classification in the Why2-Atlas tutoring system; Section 3 describes the three-tier clustering method and its experimental design, Section 4 presents the results and discussion of our experiment and Section 5 provides conclusions and directions for future work.

2 Text Classification in the Why2-Atlas System

The Why2-Atlas system presents students with qualitative physics problems and encourages them to write their answers along with detailed explanations to support their answers [8]. As shown in Fig. 1, the student explanation from our corpus of human-human computer-mediated tutoring sessions, illustrates the type of explanation the system strives to elicit from students. It is a form of self-explanation so it has the potential to lead students to construct knowledge [9], and to expose deep misconceptions [10].

Question: Suppose you are in a free-falling elevator and you hold your keys motionless right in front of your face and then let go. What will happen to them Explain.

Explanation (Essay): Free-fall means without gravity. The keys should stay right in front of your face since no force is acting on the keys to move them.

Fig. 1. An actual problem statement and student explanation

In the above example, there is a clear statement of misconception `Freefall means without gravity'. Unless we evaluate the answers that students type in, we would not be able to help them reconstruct their knowledge. There are a variety of ways in which a student essay can be evaluated or graded. Autotutor [11] uses Latent Semantic Analysis (LSA) to analyze student essays. AutoTutor comprehends the student input by segmenting the contributions into speech acts and matching the student's speech acts to the tutor's expectations. If the expectations are covered in the student's essay, the essay is considered to be `good'.

In Why2-Atlas system, we use a similar method. We first look for the correctness of the answer and then use a list of Principles (P) and Misconceptions (M) and look for the presence of a P or M in the student essay. We have an `ideal answer' for each problem statement which is clearly marked for the necessary principles to be mentioned in the respective essay. If the essay contains all of the Ps stated in the ideal answer, then it is considered to be a reasonably good essay and we allow the student to move on to the next problem. Thus, it is important to classify the students' essay strings into Ps and Ms in order to subject it to further processing.

Several other attempts have been made in the project to analyze students' essays in the past using TC methods. Rose et al.'s experiments [12] used `keypoints (correct answer aspects)' and `nothing' (in case of absence of a correct answer aspect) to classify essay strings; the precision and recall measures for the pumpkin problem[2] was 81% and 73% respectively. The limitation of this approach was the inability to generalize the training across problems and to identify misconceptions (if any) expressed by the student in his/her essay. There was an attempt to extend to more problems later in the project by identifying only `Principles' for each problem. The classifier's performance is measured in terms of accuracy and standard error. Accuracy is the percentage of correctly labeled documents in the test set. Standard error of the prediction is computed over the accuracy. The results are shown in Table 1. As the number of classes increased, the accuracy declined. Hand-checking of the tags assigned to these examples revealed many misclassifications. It was clear that the complexity of the problem lies in the nature of the data itself.

Table 1. Performance of NB classifier on subsets

Subset[3]	No. of classes	No. of examples	Accuracy	Std Error
Pumpkin	17	465	50.87	1.38
Packet	14	355	55.49	1.99
Keys	20	529	48.46	1.62
Sun	8	216	60.60	1.42
Truck	8	273	65.22	0.93

Furthermore, as this approach did not include training examples for misconceptions, the classifier grouped all such instances as `nothing' (false negatives) or put them under different `wrong' classes (false positives) neither of which was desirable by us. Since these problems share principles and misconceptions between them, yet

[2] Pumpkin was one of the 10 problems given to the students in the tutoring session.
[3] Subset includes data for the specific problems (pumpkin, keys, etc).

another approach was made to combine the examples from the subsets (in Table 1) into one. We included training examples for misconceptions as well. We tested this new dataset using the same NB algorithm and the results of this experiment are shown in Table 2:

Table 2. Performance of NB classifier on global data

Set	No. of classes	No. of examples	Accuracy	Std. Error
Global[4] (all problems)	38	586	56.83	0.45

Due to the similarity of the words present in the list of principles and misconceptions, there were still many misclassifications. To get a better understanding of the nature of the problem, we tested 15 documents that belong to one concept. We expected the classifier to tag all the documents for only one class `prin-only-gravity-implies-freefall' (The description of this principle is: "When gravity is the only force acting on an object, it is in freefall"). The classifier's predictions[5] reveal the following:

0 tagged for the expected principle `prin-only-gravity-implies-freefall' (Class1)
12 tagged for `prin-drop-obj-only-grav' (Class2)
1 tagged for `prin-release-freefall' (Class3)
4 tagged for `prin-freefall-same-accel' (Class4)
1 tagged for `nothing' (Class5)

Based on the training data, the classifier thus, encountered different but related principles for the above set of data. This led us to examine the probability distribution of the words used by each of these classes. Table 3 shows the probability distribution of the top 10 words.

It can be observed that the significant words `gravity, free and fall' are found in all the classes (2–4) and hence the problem of ambiguity arose. However, it should be noted that the tags obtained above are related to each other. One can say that they are partially correct and are related to the top principle in question. For instance, `*prin-drop-obj-only-grav*' is a subset of `*prin-only-gravity-implies-freefall'*. So, based on the combined probability of the key words that are common for both these principles, the classifier learned `prin-drop-obj-only-grav' as in *"*The only force acting on the man and the keys is the force of gravity*"*. Later on, we tested a few more sentences chosen randomly that contained words like `freefall' and `gravity'. Hand-checking of the predictions revealed that a sentence like `Freefall means without gravity' (a misconception) was classified as a principle. This is not surprising because `without' was

[4] This included data from all the ten problems.

[5] The mismatch in the number of tags (18) and the number of sentences (15) is due to some segmentation problems. Some of the documents were broken into more than one due to the presence of a `period'. The principles corresponding to Classes 2, 3, and 4 are related to the concept of `freefall' but do not correspond to the exact description of the concept in Class1.

Table 3. Info-gain of the top 10 words using NB

Class2		Class3		Class4	
words	**probability**	**words**	**probability**	**words**	**probability**
force	0.056206	force	0.021341	keys	0.027356
gravity	0.046838	free	0.018293	freefall	0.024316
keys	0.039813	fall	0.018293	elevator	0.024316
acting	0.035129	gravity	0.015244	free	0.018237
elevator	0.023419	acting	0.015244	person	0.015198
fall	0.018735	keys	0.015244	fall	0.015198
free	0.014052	freefall	0.012195	release	0.012158
rate	0.011710	acceleration	0.009146	accelerating	0.006079
accelerating	0.009368	elevator	0.009146	sentence	0.006079
front	0.009368	problem	0.009146	previous	0.006079

listed as a stop word (whose `info-gain' is lower than the rest of the words) in our experiment. So we decided `not to' use a stop-list in our future experiments. But, still this would not solve the problem completely because the naïve bayes algorithm ignores the relationships of significant words that do not co-occur in the document. Hence, we investigated the performance of various other classifiers on this issue and decided to use k-nearest neighborhood algorithm along with the new clustering technique[6].

3 Experimental Design

In this section, we describe our new experiment, the datasets used in the experiment and the coding scheme at length.

3.1 Dataset

All of the datasets used in this work are extracted from the WHY-Essay[7] corpus that contains 1954 sentences (of essays). A list of Principles and Misconceptions that corresponds to physics concepts of interest in the Why2-Atlas project is used as the set of classes to be assigned to these essay strings. There are 50 principles and 53 misconceptions in total.

The training and test data are representative samples of responses to physics problems drawn from the same corpus. We created tag-sets for both principles and misconceptions (a total of 103) and used these to tag the data. We carried out many trials

[6] For reasons of space, the statistical results of the various other classifiers used for this purpose are not shown here.

[7] The Why-essay corpus consists of students' essay statements mostly from Spring and Fall 2002 experiments of human-human tutoring sessions.

of classification and the performance on `old data' was used to do data-cleaning and to revise the relations between classes that we want to identify/predict. Due to scarcity of quality-data of essays containing misconceptions, we had to write some student-like statements in order to expand the corpus of training data for misconceptions. This required human expertise and a good understanding of the subject matter.

3.2. Creation of Clusters

The Principles and Misconceptions used for tagging the essay segments have similar topics (e.g. gravity-freefall and gravitational force, second law etc) and therefore share common words. The classification task is typically hard because of lack of unique terms and thus increases the feature dimensionality of these documents. Thus, it is highly desirable to reduce this space to improve the classification accuracy. The standard approach used for this kind of task is to extract a `feature subset' of single words through some kind of scoring measures (for example, using `Info-gain'). The basic idea here is to assign a score to each feature (assigned to each word that occurred in the document), sort these scores, and select a pre-defined number of the best features to form the solution feature subset (as in Latent Semantic Indexing approaches). In contrast to this standard approach, we use a method to reduce the feature dimensionality by grouping "similar" words belonging to specific concepts into a smaller number of `word-clusters' and viewing these `clusters' as features. Thus, we reduce the number of features from `hundreds' to `tens'.

Though there have been many studies (for example, [13]) that use word-clusters to improve the accuracy of unsupervised document classification, there are very few studies that have used this kind of indirect `supervised' clustering techniques for text classification. Baker and McCallum [14] showed that word-clustering reduced the feature dimensionality with a small change in classification performance. Slonim and Tishby [4] use an information-bottleneck method to find word-clusters that preserve the information about the document categories and use these clusters as features for classification. They claim that their method showed 18% improvement over the performance of using words directly (given a small training set). Our work is unique in that it uses a three-tier word-clustering method to label each student essay statement. We endorse the same claims as the other two works, that word-clustering even when done on classes instead of directly on the data improves the classification performance significantly.

3.2.1 The Three-Tier Clustering Method

Determining the `similarity' of words in these physics documents is a difficult task. Given the list of the principles and misconceptions used for tagging the students' essay strings, we examined the semantics of the descriptions of each principle and misconception and extracted those words (word clusters) that seemed to best describe a particular concept and put them together. Fig. 2 illustrates this idea.

Thus, we have a three-tier tagging schemata that we built by hand in a bottom-up fashion:

cluster, sub-cluster and class

The upper levels (cluster and sub-cluster) describe the topic of discussion and the lower level describes the specific principle or misconception. The + sign in each node means the presence of that particular `word(s)' in a concept description. For example,

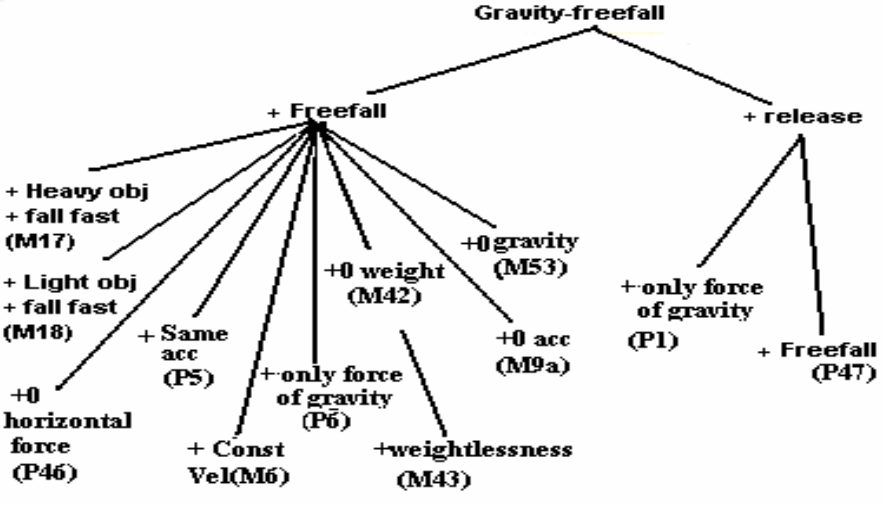

Fig. 2. Chart showing the features related to the cluster `Gravity-Freefall'

from the trees in Fig 2, we can see that **+freefall** and **+only force of gravity** describe Principle `P6' while **+freefall** and **+0gravity** describe a Misconception `M53'. Thus, words in the lower level that are shared across concepts migrate into an upper tier. The top-most level was created using the concepts described at the middle level. We created ten such clusters based on the prominent keywords for the training data (see Table 4 for specifics[8]).

This information was used to extend the original corpus annotations[9] so that the training data took the form (mapping each D to C):

C = {(clustername, subclustername, class)

as exemplified below:

Freefall acceleration is the same for all objects and the keys the person and the elevator are all accelerating downwards with the same acceleration. {*gravity-freefall, freefall-prin, prin-freefall-same-accel*}.

If the two forces were equal they would cancel each other out because they are in opposite directions. {*3rdLaw, act-react, misc-act-react-cancel*}

In addition, there was also a `nothing' class. The student statements that were neither a `P' nor a `M' are in this class.

[8] Absence of sub-clusters in some groups means that there was no ambiguity between the principles and misconceptions in that cluster. The numbers found under the third column indicate the number of principles and misconceptions that fall under the respective sub-cluster.

[9] Annotations were for the principles, misconceptions and `nothing'.

Table 4. The three-tier clusters of principles and misconceptions

Cluster	Subcluster	Classes	
		P	M
Gravity-freefall	Freefall	3	7
	Release	2	0
Gravita-tional-force	-	3	11
Secondlaw	Netforce	3	1
	Force	2	8
Thirdlaw	One-object	1	0
	2obj	4	1
	Act-react	0	5
Kinematics and vectors	Force	2	1
	Zero- netforce	4	0
One-object-second-third-law	Lightobj	0	4
	heavyobj	0	2
	objhit	0	1
Two-objects-motion	Samevel	7	2
	cons.vel- over-t	1	0
	jointmotion	3	0
Acceleration-velocity-displacement	-	4	1
Weight-mass	-	0	4
General	-	5	5

3.3 Document Modeling

Our main interest is to prove that `BOW approach with clusters' outperforms `BOW approach without clusters' on students' essay strings. Additionally, we are concerned with how this comparison is affected by the size and the nature of the training set. In this section, we discuss the various stages of our `document modeling'.

Document Indexing

Texts cannot be directly interpreted by a classifier or by a classifier-building algorithm. Therefore, it is necessary to have an indexing procedure that maps a text (dj) into a compact representation of its content. This should be uniformly applied to training, validation, and test documents. We used the bag-of-words representation to index our documents with binary weights (1 denoting presence and 0 absence of the term in the document). A document for us is a whole proposition and not a general topic (commonly used in most BOW approaches to classify web pages).

The `k- Nearest Neighborhood (kNN) Classifier
We used a simple k-Nearest Neighborhood (kNN) algorithm[10,] which is an instance based learning approach for the classification task. Fix and Hodges defined a metric to measure "closeness" between any two points and formulated a kNN rule: `Given new point x and a training set of classified points, compute the kNN to x in the training data. Classify x as Class y if more k-nearest neighbors are in class y than any other class' [15]. In the context of TC, the same rule is applied where documents are represented as `points' (vectors with term weights). We used the Euclidean distance formula to compute the measure of "nearness'.

Procedure
kNN was used for the three-tier clustering model that included the following stages:

1. Modeling the dataset (X) at the cluster level,
2. Dividing the dataset (X) into sub-datasets (Y) for sub-clusters, and bifurcating them into two (one for principles and another for misconceptions)
3. Modeling the sub-datasets (Y) at the subcluster level
4. Dividing the dataset (X) into sub-datasets (Z) for the third level (classes),
5. Modeling the subdatasets (Z) at the class level.

The classification outputs at each level were the cluster, subcluster and class tags respectively. At runtime, the output of a level is used to select a model in the next level.

Cross-Validation
We used the 2/3 and 1/3 split of training and test data for this experiment. We set the value of `k' to 1 in kNN and evaluated kNN on the test set. A stratified ten-fold cross-validation was repeated over 10 trials to get a reliable error estimate.

4 Results and Discussions

The metrics used to measure the performance of the learner are: accuracy, standard error, and precision and recall. In order to define precision and recall, we need to define `true positives, false positives, and false negatives. In our context, if a document D is related to C, it will be considered to be a `True Positive (TP)', with a value of `1'. If a document D is not related to C, it will have a value of `0' and can either be marked as `nothing' which constitutes the `False Negatives (FN)' for us or it can be misclassified (as some other C) which means that it is a `False Positive (FP)'. For example, if a student string `Freefall means without gravity, is correctly classified as misconception statement (M53), it is a TP. On the other hand, if it is categorized as `nothing' then it is a `FN' and if it is misclassified as anything else then it is `FP'. Precision and Recall can thus be defined as:

[10] We used the kNN algorithm from the RAINBOW software devised by McCallum (1996). McCallum, Andrew Kachites. "Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering", www.cs.cmu.edu/~mccallum/bow.

Precision = TP/ (TP + FP),
Recall = TP/ (TP + FN).

Using this formula, we computed the recall and precision of the bow-approach with three-tier clusters (see Table 5):

Table 5. Precision and Recall results[11] for the three-tier model

Model		Precision (%)	Recall (%)
Three-tier clustering	Cluster(one level)	80.88	92.13
	Subcluster(two levels)	74.25	88.75
	Classes (three levels)	62.58	90.75
Without clustering (using NB)		68.59	83.41

The accuracy and standard error of prediction at each level of clustering are shown in Table 6 below along with the statistics of the bow-only approach using naïve bayes classifier without clustering:

Table 6. Accuracy and Standard Error of the three-tier model

Model		Accuracy	Std. Error
Three-tier clustering	Cluster (one level)	78.01	0.016
	Subcluster(two levels)	74.50	0.020
	Classes(three levels)	64.16	0.185
Without clustering (using NB)		50.99	0.019

The above results show that the three-tier clustering indeed helped to improve the performance of the classification. Ambiguity (or noise) among classes was significantly reduced as the documents were forced to traverse the whole path (cluster → subclusters → classes). Our model significantly outperformed the bow-only approach using the naïve bayes classifier (27.02%, 23.51% and 13.17% of improvement in the classification accuracy for the levels 1, 2 and 3 respectively).

5 Conclusions and Future Directions

This paper discussed a three-tier clustering approach of classifying data pertaining to students' essay statements of qualitative physics problems in a tutoring system. We claim that `supervised three-tier clustering' outperforms the non-clustering models related to this domain. We conjecture that expansion of the training corpus for more examples for misconceptions will further improve the clustering results and thereby aid us in effective evaluation of the students' essays.

[11] The measures at each level use the output of the previous level regardless of correctness.

Acknowledgements. This work was funded by NSF grant 9720359 and ONR grant N00014-00-1-0600.

References

1. Maron, M. Automatic indexing: an experimental inquiry. *Journal of the Association for Computing Machinery* (1961) Vol. 8(3): 404–417.
2. Duda, R., and Hart, P. Pattern Classification and Scene Analysis. John Wiley & Sons. (1973) 95-99.
3. Sebastiani, Fabrizio. Machine learning in automated text categorization. *ACM Computing Surveys,* (2002) Vol.34(1):1–47.
4. Slonim, N. and Tishby, N. The power of word clusters for text classification. In *Proceedings of ECIR-01, 23rd European Colloquium on Information Retrieval Research*, Darmstadt, Germany, (2001).
5. Slonim. N. N. Friedman, and N. Tishby. Unsupervised document classification using sequential information maximization. *Proceedings of SIGIR'02, 25th ACM international Conference on Research and Development of Information Retireval,* Tampere, Finland, ACM Press, New York (2002).
6. El-Yaniv, R and Oren Souroujon. Iterative Double Clustering for Unsupervised and Semi-supervised Learning. European Conference on Machine Learning (ECML) (2001) 121-132.
7. Periera, F, N. Tishby, and L. Lee. Distributional clustering of English words. In *31st Annual Meeting of the ACL*, (1993) 183-190.
8. VanLehn, Kurt, Pamela Jordan, Carolyn Rose´, Dumisizwe Bhembe, Michael Bottner, Andy Gaydos, Maxim Makatchev, Umarani Pappuswamy, Michael Ringenberg, Antonio Roque, Stephanie Siler, and Ramesh Srivastava. The architecture of Why2-Atlas: A coach for qualitative physics essay writing. In *Proceedings of Intelligent Tutoring Systems Conference, volume 2363 of LNCS*, Springer. (2002) 158–167.
9. Chi, Michelene, Nicholas de Leeuw, Mei-Hung Chiu, and Christian LaVancher. Eliciting self explanations improves understanding. *Cognitive Science*, (1994) Vol. 18:439–477.
10. Slotta, James, Michelene T.H. Chi, and Elana Joram. Assessing students' misclassifications of physics concepts: An ontological basis for conceptual change. *Cognition and Instruction*, (1995) Vol. 13(3):373–400.
11. Graesser, A. C.; Wiemer-Hastings, P.; Wiemer-Hastings, K.; Harter, D.; Person, N.; and the Tutoring Research Group. Using Latent Semantic Analysis to Evaluate the Contributions of Students in AUTOTUTOR. *Interactive Learning Environments* (2000) Vol. 8:129–148.
12. Rosé C,P, A. Roque, D. Bhembe, K. VanLehn. A Hybrid Text Classification Approach for Analysis of Student Essays, Proceedings of the Human Language Technology conference/ North American chapter of the Association for Computational Linguistics annual meeting. *Workshop on Educational Applications of Natural Language Processing*. (2003).
13. Hotho, Andreas, Steffen Staab and Gerd Stumme. Text Clustering Based on Background Knowledge. Institute of Applied Informatics and Formal Description Methods AIFB, *Technical Report No. 425.* (2003).
14. Baker, L.D. and A. K. McCallum. Distributional Clustering of Words for Text Classification. In ACM SIGIR 98. (1998).
15. Fix, E. and J.L. Hodges. Discriminatory Analysis -- Nonparametric Discrimination: Consistency Properties, Project 21--49--004, Report No. 4, USAF School of Aviation Medicine, Randolf Field, TX (1951) 261--279.