Association for Information Systems
AIS Electronic Library (AISeL)

ICEB 2002 Proceedings | International Conference on Electronic Business (ICEB)

Winter 12-10-2002

A Feature Weighting Method by Multimedia Data Model on E-Business

Young-Jun Kim

A Feature Weighting Method by Multimedia Data Model on E-Business

Young-Jun Kim
Department of Secretarial Information Studies
Cheonan College of Foreign Studies
Cheonan, Korea
yjkim@ccfs.ac.kr

Abstract

A lazy learning method has relative advantages in comparison to eager learning method. However lazy learning has relative disadvantages also. Lazy learners are sensitive to irrelevant features. When there are irrelevant features, lazy learners have difficulty to compare cases. This is one of the most critical problems and the accuracy of reasoning can be degraded significantly. To overcome this restriction, feature weighting method for lazy learning have been studied. All the methods previously proposed tried to improve some parts of this generic process with different approaches. However, most of the existing researches were focused on global feature weighting. Therefore, we propose a new local method on e-business. The motivation to try local feature weighting method is that there are situations where locally varying weight vectors can help improving classifier performance by multimedia data model on e-business.

1. Introduction

By using a set of previously encountered cases, each of which typically represented by a set of features, classification methods attempt to produce class descriptions that will be accurate for new cases. The class assigned to a new case can then be used to decide how to process it. There are two types of learning modes: eager learning and lazy learning. Eager learning approaches to induction produce generalizations that explicitly represent the classes under study, often in a language different from that used to represent the cases. Lazy approaches, in contrast, delay this generalization process until classification time; it is performed implicitly when a new case is compared to the stored cases and the class of the nearest one(s) is assigned to it.

There will be many potential advantages if we automate the feature the feature weighting process. Caruana and Freitag[6] described the advantages of feature selection. By adapting their suggestion, we can describe the advantages of automated feature weighting as showed follows: It gives the learning system designer freedom to identify as many potentially useful features as possible and then let the learning system automatically determine which ones get heavier weight and which ones get lighter weight. It allows new features to be added easily to a learning system. It allows the weight of features to change dynamically as the amount of training data changes on e-business.

2. Backgrounds of Study

The field of machine learning was conceived nearly four decades ago with the objective to develop intelligent computational methods that would implement various forms of learning, in particular mechanisms capable of inducing knowledge from example or data. One of the vital invention of artificial intelligence(AI) research is the idea that formally intractable problems can be solved by extending the traditional scheme program = algorithm + data to the more elaborate program = algorithm + data + domain knowledge.

As seen in the above equation, applying the domain knowledge is fundamental for solving problems in the field of AI. However, the use of knowledge does nothing but shifts bottleneck of implementing the AI program from the programmer to the knowledge engineer. In other words, the process of knowledge acquisition and encoding is still far from being easy. Thus a tempting idea springs to mind: employ a learning system that will acquire such high-level concepts and/or problem-solving strategies through examples in a way analogical to human learning. Most research in machine learning has been devoted to developing effective methods to address this problem.

CBR is one of such machine learning approaches. Previous cases are used to make a solution for a new problem. From the cases available, a CBR system retrieves the most similar case(s) to the input problem and then adapts the solution of the retrieved case to the fit the context of the Input problem. The basic idea of CBR is based on the process of human problem solving. Human beings use previous experiences of problem solving when encountered a new problem to solve. This natural problem solving approach allows the reuse of problem solving experiences and is considered a breakthrough from the knowledge acquisition bottleneck in the artificial intelligence area.

A process model of Riesbeck and Schank has been a popular and most widely used CBR process model[14]. The CBR model has six major stages: indexing, retrieval,

The Second International Conference on Electronic Business
Taipei, Taiwan, December 10-13, 2002

adaptation, test, indexing and store, explanation and repair, The process of CBR also requires stored knowledge structures: case base, indexing rules, similarity matrix or metrics, adaptation rules, repair rules. The case base stores the cases previously solved and the indexing rules help searching most similar and useful cases efficiently and effectively. The similarity metrics are used to calculate the similarity or distance of a new case from a case stored in the case base and the repair rules are used in correcting failed solutions proposed by the CBR process.

In this study, we focus on the typical classification problems that have the following characteristics. The problems have discrete output classes. Hence, the performance of CBR system can be investigated by checking the results whether they are correct or not. The problems have relatively many features and have both numeric and categorical features in most cases.

3. Categorization of FW Methods

Feature weighting(FW) efforts attempt to find the optimal feature weight vector that makes the classifier show best classification accuracy. Feature weighting methods search through the feature weight vectors, and try to find the best one among the unlimited number of candidate weight vectors according to an evaluation criterion. However, this procedure is exhaustive because it tries to find only the best one. It may be too costly and practically prohibitive, even for a small size of feature set. Other methods based on heuristic or random search attempt to reduce computational complexity at the cost of performance. These methods need a stopping criterion to prevent an exhaustive search of weight vectors. There are four basic steps in a typical feature weighting method. (1) A generation procedure to generate candidate weight vectors. (2) An evaluation procedure to evaluate the weight vector examination. (3) A stopping criterion to decide when to stop and (4) A validation procedure to check whether the weight.

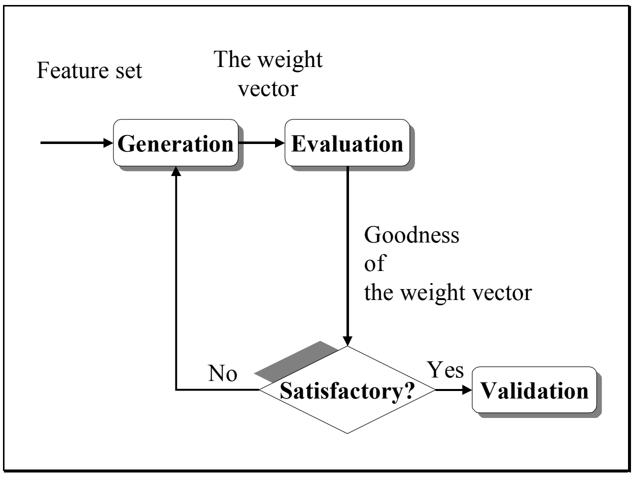

Figure 1 A generic feature weighting process

Figure 1 shows a generic feature weighting process methods are several frameworks for categorizing feature weighting, specifically, feature selection methods[7], we will use Dash and Liu's framework as a base for our framework to Include local feature weighting methods. Dash and Liu suggested a 2-dimensional categorization framework of feature selection methods. Although their framework is of feature selection methods, there is no significant difference to the framework of feature weighting. This is because feature selection is a special case of feature weighting as we mentioned in backgrounds. Their framework considered generation procedures and evaluation functions as the most critical dimensions. Each feature selection method is grouped depending on the type of generation procedure and evaluation function used. They chose 32 methods and then grouped them according to the combination of generation procedure and evaluation function used.

However, they did not consider the dimension of the scope of weight. Hence, we add the scope dimension and present a 3-dimensional framework in order to classify local and global feature weighting methods. Table 1 shows the modified framework and categorization of some representative methods.

Table 1 Categorization of feature weighting methods in a 3-dimentional framework

Scope of Weight	Evaluation Function	Generation Procedure		
		Heuristic	Complete	Random
Global	Distance	Relief	+	-
	Information	DTI	+	-
	Dependency	+	-	-
	Consistency	-	+	+
	Classifier Error	+	+	CA
Local	Classifier Error	RC	-	This Study

+ There are several methods but they are not presented here.

- There are no known methods.

Relief and decision tree induction(DTI)[11] use heuristic generation procedures. GA generates feature weight vectors randomly. RC and this study, the feature weighting method we develop in this research, are local and wrapper feature weighting methods. RC uses heuristic generation procedure and this study uses random generation procedure.

4. Sequential Weighting Algorithms

4.1 Forward and Backward Algorithm

The generation procedure generates the feature weight values that will be evaluated by an evaluation procedure. The generation procedure can start with (1) all 0-weight values, (2) all 1-weight values or (3) randomly generated weight values. Methods that have property of (1) are called FSS methods, whereas methods that have property of (2) are called BSS methods. In (1), feature weights are iteratively increased until no further improvement is possible. In (2), feature weights are iteratively decreased. In (3), feature weights are randomly generated in each iteration.

4.2 Relief Algorithm

Relief algorithm uses a statistical method to weight the relevant features. From the set of training cases, it first chooses a sample of cases, where the number of samples are provided by the user Relief randomly picks this sample of cases, and for each case it finds near Hit and near Miss cases based on Euclidean distance measure. Near Hit is the case having minimum Euclidean distance among all the cases in the same class as that of the chosen case; near Miss is the case having minimum Euclidean distance among all the cases in the different class. The initial values of feature weights were set to zero in the beginning. Relief updates the feature weight using the information obtained from near Hit and near Miss. A feature is more relevant if it differentiates a case from its near Miss, and less relevant if it differentiates a case from its near Hit. After exhausting all cases in the sample, it selects the features whose weights are greater than or equal to a threshold. Relief works for noisy and correlated features, and requires only linear time with respect to the number of given features and number of samples. A limitation is that it does not detect redundant features. Another limitation is that the user may find it difficult to provide a proper number of samples.

5. Feature Weighting Procedures

5.1 using Decision Tree Induction

Decision tree(DT)-based feature weighting methods use heuristic weight generation and information gain as evaluation measure. Cardie[11] showed that the use of feature weighting generated by DT can improve the performance of CBR. DT generation method such as C4.5[12] is run over the training set, and the features that appear in the pruned DT are selected. Some variations are also possible. For example, after generating a DT, original features can be weighted according to the entropy values[5].

5.2 using Genetic Algorithm

There are several approaches using genetic algorithms (GA) for weighting features. The average classification accuracy of GA-kNN was almost 81%, which is very high in the sense that the accuracies of basic CBR models were approximately 63%. However, feature weighting methods using GA need to assign proper values to such parameters as maximum number of iterations, initial population size crossover rate and mutation rate.

6. Evaluation Procedures of Study

Evaluation procedures can be categorized differently by whether they use feedback from the performance task. Methods that do not use feedback are called filters, whereas methods that use the classifier itself as the evaluation procedure are wrappers[10]. Since the features are selected using the classifier that later on uses these selected features in predicting the class of unseen cases, the accuracy of wrapper model is high. Some evidences suggest that wrapper models are superior to filter models (e.g., Wettschereck et al.[16]) when the dependent variable is classification accuracy. However, wrapper models are often more computationally expensive[7]. Filter models and other efficient variants of wrapper models should be considered for tasks when computational expense is a critical concern. Figure 2 and figure 3 show a generic process of wrapper model and filter model, respectively, presented by John et al.[10].

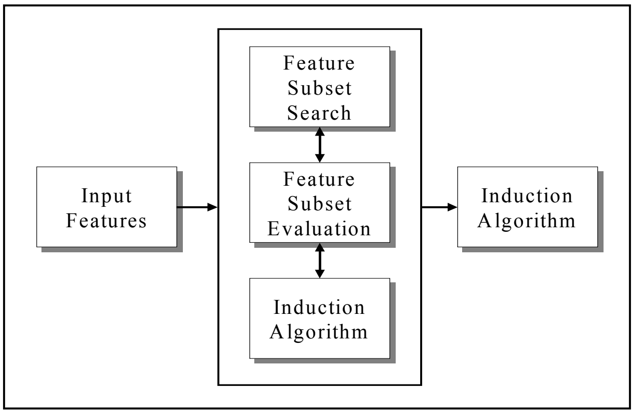

Figure 2 Process of a generic wrapper model

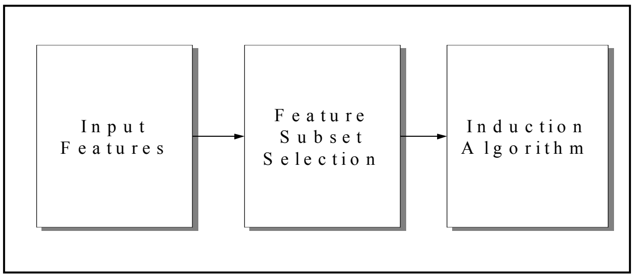

Figure 3 Process of a generic filter model

7. Local Feature Weighting Methods

The scope of weights means the generality of the weights in the case space. The scope of weights that most feature weighting methods produce is global: their weights apply across the entire case space. In contrast, local feature weighting methods allow feature weights to vary in different parts of the case space. In local feature weighting methods, weights can vary by class, feature value, and/or individual case or subset of cases.

Domingos' RC algorithm uses a case-specific feature weighting algorithm. RC is in many ways similar to BSS, but it makes case-specific decisions on feature relevance.

It drops features from a case if (1) their values differ from the case's nearest neighbour and (2) removing them does not decrease overall leave-one-out-cross-validation error(LOOCE). After removing features from the original case base, duplicate cases may be produced, but are not removed.

In a setting with only single nearest neighbour, RC outperformed both FSS and BSS significantly on 24 data sets, and showed increased efficiency with increasing context-dependency of feature relevance. However, RC is limited to binary weights and it is hard to extend RC to allow continuous weights.

8. Summaries and Discussion

There are some researches on flexible, context-sensitive, and local feature weighting. However, few researches tried to use wrapper model for local feature weighting except for Domingos. We propose a new local wrapper method for feature weighting. This system is very simple and relatively efficient among wrapper model-based feature weighting methods. Our methods overcome the limitations Of RC and this study will support more than single nearest neighbour. We can enhance the classification performance by multimedia data model on e-business.

Although the results of some applications did not showed sufficient evidences for the usefulness of the new method, we expect that it can be improved and will work effectively in most situations. That is because the core idea of the method is remembering the real experiences.

The core contribution of this research can be stated as: (1) We will extend existing categorizations of feature weighting methods by including the scope dimension, and develop a new 3-dimensional framework and then develop a brand new combination of feature weighting method on e-business. (2) We will develop a new measurement called input dependency of feature relevance that will be used to determine which type of weights, i.e., local weights or global weights, is appropriate for a particular application.

References

[1] Aha, D. W., "Feature Weighting for Lazy Learning Algorithms," Liu, H. and H. Motoda(Eds.), *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Norwell MA: Kluwer, 1998.

[2] Berry, M. J. and G. Linoff*, Data Mining Techniques for Marketing, Sales, and Customer Support*, John Wiley and Sons, 2001.

[3] Blake, C., E. Keogh, and C. J. Merz, *UCI Repository of machine loaming databases*, Irvine, CA: Univ. of California, Department of Information and Computer Science, 1998.

[4] Cardie, C., "Using Decision Trees to Improve Case-Based Reasoning," *Proceedings of the 10th International Conference on Machine Learning*, Morgan Kaufman, 1997. pp.25-32.

[5] Cardie, C. and N. Howe, "Improving Minority Class Prediction Using Case-Specific Feature Weights,". *Proceedings of 14th International Conference on Machine Learning*, 2000. pp.57-65.

[6] Caruana, R. and A. D. Freitag, "Greedy Attribute Selection," *Proceedings of the Eleventh International Conference on Machine Learning*, 1998.

[7] Dash, M. B. and A. H. Liu, "Feature Selection for Classification," *Intelligent Data Analysis*, Vol.3 No.3, 2001.

[8] Domingos, P., "Context-Sensitive Feature Selection for Lazy Learners," *Artificial Intelligence Review*, Vol.11, 2001. pp.227-253.

[9] Howe, N. and B. C. Cardie, "Examining Locally Varying Weights for Nearest Neighbor Algorithms," Case-Based Reasoning Research and Development: *Second International Conference on Case-Based Reasoning,* 1997. pp.445-466.

[10] John, G. H., R. Kohavi and K. Pfleger, "Irrelevant Features and the Subset Selection Problem," *Proceedings of the Eleventh International Conference on Machine Learning*, 1998. pp.121-129.

[11] Kira A. and L. A. Rendell, "A Practical Approach to Feature Selection," *Proceedings of the 9th International Workshop on Machine Learning*, 1996. pp.249-256.

[12] Michalski, R. S., I. Bratko, and M. Kubat, *Machine Learning and Data Mining: Methods and Applications*, John Wiley & Sons, 2001.

[13] Nelson, M. M., and W T. Illingworth, *A Practical Guide to Neural Nets*, Addison-Wesley, 1991.

[14] Riesbeck, C. K. and R. L. Schank, *Inside Case-Based Reasoning*, Lawrence Erlbaum Associates, 1995.

[15] Quinlan, J, R., C4.5: *Programs for Machine Learning, San Mateo*, CA: Morgan Kaufman, 1993.

[16] Wettschereck, D., D. W. Aha, and T. Mohri, "A Review and Empirical Comparison of Feature Weighting Methods for a Class of Lazy Learning Algorithms," *AI Review*, Vol.11, 2001. pp.273-314.