The Principle of Transformation between Efficiency and Effectiveness: Towards a Fair Evaluation of the Cost-Effectiveness of KDD Techniques

Alex A. Freitas[1]
University of Essex, Dept. of Computer Science
Wivenhoe Park, Colchester, CO4 3SQ, UK
freial@essex.ac.uk
http://cswww.essex.ac.uk/projects/res/freial/web/alex.html

Abstract

Most of the KDD literature focuses on analyzing the effectiveness of KDD techniques, in the sense e.g. of reducing the classification error rate in the case of classification tasks. Efficiency issues are usually considered of secondary importance, it considered at all. In contrast, we focus on the cost-effectiveness of KDD techniques, i.e. on the trade-off between effectiveness (reduction of error rate) and efficiency (reduction of processing time). In particular, we show that a gain in efficiency can be transformed into a gain in effectiveness, and this principle can be used to evaluate the cost-effectiveness of KDD systems in a fair manner. We discuss the application of this general principle to evaluate the cost-effectiveness of two general kinds of KDD techniques, namely classification algorithms and attribute selection algorithms.

1 Introduction.

KDD techniques can be evaluated along several dimensions, such as effectiveness (e.g. some measure of error rate), efficiency (related to processing time), comprehensibility of the discovered knowledge, etc. In this paper we are interested in two of these evaluation dimensions, namely effectiveness and efficiency.

Most of the literature focuses on analyzing the effectiveness of KDD techniques. The analysis of the efficiency of KDD techniques, if it is done at all, tends to be done in a way separate from the analysis of effectiveness. In other words, effectiveness and efficiency are usually considered two very different dimensions, and the trade-off between these two dimensions has been little investigated in the KDD literature. This approach can be regarded as a "reductionist" one, in the sense that effectiveness and efficiency are studied separately from each other.

In contrast, this paper follows a more "holistic" approach to cope with this trade-off. More precisely, this paper focus on the cost-effectiveness of KDD techniques, i.e. on the trade-off between effectiveness (measured by the minimization of error rate) and efficiency (measured by the minimization of processing time). To simplify our discussion, throughout this paper we assume that the target task is classification, by far the most addressed task in the literature. However, most of the arguments presented in this paper can be generalized to other KDD tasks.

[1] Financially supported by the Brazilian government's CNPq, grant No. 200384/93-7.

The motivation for the ideas presented in this paper can be explained as follows. Suppose we want to compare two KDD techniques to solve a given problem. For instance, we could compare two classification algorithms or two attribute selection algorithms used as pre-processing for classification. Clearly, if one of the two techniques being compared is both more effective and more efficient than the other, then the former can be definitely considered superior to the latter, along the dimensions of effectiveness and efficiency.

In practice, however, often one technique, say technique A, is more effective but less efficient then the other, say technique B. The question this paper aims to answer is: How can we compare the *cost-effectiveness* of two KDD techniques aimed at solving a given problem, when technique A is more effective but less efficient than technique B? Before we start to answer this question, we justify its importance.

First of all, it should be noted that this question addresses a ubiquitous situation in KDD. Intuitively, the more time an agent (in our case, a KDD technique) spends to solve a given problem, the better the quality of the problem solution that the agent will find. Hence, if technique A is less efficient (i.e. takes more time) than technique B, it is reasonable to expect that A is more effective in solving the target problem than B.

This intuition is not always true. There are exceptions. Consider, for instance, the case of two versions of a classification algorithm doing a different amount of search. Contrary to our intuition, [Quinlan & Cameron-Jones 95] showed that increasing the amount of search (i.e. allocating more processing time to the algorithm) can lead to an increase in the error rate (making the algorithm less effective).

However, despite the existence of exceptions, the above described intuition remains true in many practical situations. Actually, sometimes we can make sure that an increasing in the processing time allocated to a KDD technique has the effect of monotonically increasing its effectiveness. For instance, we can run a classification algorithm several times, each time with a different setting of parameters, and pick up the solution with the smallest error rate (i.e. the most effective) of all these runs.

To summarize, in KDD research the situation where a given technique A is more effective but less efficient than a given technique B is commonplace. This paper proposes a fair way of evaluating the cost-effectiveness of two KDD techniques in this commonplace situation. As will be seen later, the proposed solution is based on the general principle of transformation between efficiency and effectiveness.

This paper is organized as follows. Section 2 discusses how we can evaluate the cost-effectiveness of KDD techniques, and proposes a solution based on the principle of transformation between efficiency and effectiveness. Section 3 shows how this principle can be applied to evaluate the cost-effectiveness of classification algorithms. Section 4 shows how this principle can be applied to evaluate the cost-effectiveness of attribute selection algorithms. Finally, Section 5 presents the conclusions.

2 Comparing the Cost-Effectiveness of KDD Techniques.

Let E_A denote the error rate achieved by using a given technique A, and let T_A denote the processing time taken by the technique A. Similarly, let E_B and T_B denote respectively the error rate and the processing time associated with a given technique B. As discussed in the Introduction, we are particularly interested in the commonplace situation where $E_A < E_B$ and $T_A >> T_B$.

A straightforward way of analyzing the cost-effectiveness (i.e. the trade-off between effectiveness and efficiency) of a KDD technique consists of calculating a combined measure (e.g. a weighted average) of its error rate and its processing time. Obviously, we cannot directly add up an error rate value to a processing time value, since the resulting sum would be meaningless, due to the difference of dimensional units. However, we could somehow normalize both error rate and processing time.

For instance, the processing time of the slowest technique (T_A) can be normalized to the value 1, while the processing time of the fastest technique (T_B) can be normalized to a value in the range [0..1], as given by T_B/T_A. Similarly, E_B can be normalized to 1, while E_A can be normalized to a value in the range [0..1], as given by E_A/E_B. Hence, we could try to come up with a single formula involving two variables, namely normalized error rate and normalized processing time, where each of these variables would have a numerical weight indicating its importance in the evaluation of the overall performance. For instance, assigning the weight 0.7 to error rate and the weight 0.3 to processing time, the cost-effectiveness of a given KDD technique A and a given KDD technique B, respectively denoted CE_A and CE_B, would be measured by:

$$CE_A = 0.7\ E_A/E_B + 0.3\ , \qquad CE_B = 0.7 + 0.3\ T_B/T_A\ .$$

Hence, the most cost-effective attribute selection technique would be the one having the smallest CE, as measured by the above formulas. Although this kind of normalization renders the CE measure more meaningful than its non-normalized counterpart, this approach still has an inherent problem. It would be difficult to determine "good" values for the numerical weights, since the relative importance of error rate and processing time is highly domain-dependent and, of course, very subjective. The next Subsection proposes an alternative, objective way to measure the cost-effectiveness of a KDD technique.

2.1 The Principle of Transformation between Efficiency and Effectiveness.

This Subsection proposes a fair, objective way to evaluate the cost-effectiveness of KDD techniques. The motivation for the proposed solution can be understood by an analogy with the well-known trade-off between the predictive accuracy and the (syntactic) complexity of a theory. An elegant way of coping with this trade-off is to use the Minimum Description Length Principle (MDLP) [Quinlan 94], [Quinlan & Rivest 89]. According to the MDLP, the best theory is the one requiring the smallest number of bits to encode both the theory and the data given the theory. Hence, the MDLP principle allows us to analyze both the predictive accuracy and the complexity of a theory in the same dimensional unit, namely bits.

Since we are interested in analyzing the trade-off between effectiveness and efficiency, what we need is a principle analogous to the MDLP that allows us to measure both effectiveness and efficiency in terms of a common dimensional unit.

Hence, in order to analyze the cost-effectiveness of a KDD technique we propose the following general principle of transformation between efficiency and effectiveness: *a gain in efficiency can be transformed into a gain in effectiveness*. As a result, we can analyze cost-effectiveness in terms of the dimensional unit used to analyze effectiveness, often the error rate.

To show how this principle can be actually applied, and to show the generality of this principle, we will show how to apply it to evaluate the cost-effectiveness of two quite different kinds of KDD techniques, namely classification algorithms and attribute selection algorithms. This will be shown in Sections 3 and 4, respectively.

3 Applying the Principle of Transformation Between Efficiency and Effectiveness to Classification Algorithms.

It is often the case that two classification algorithms, when applied to the same database, achieve significantly different error rates and take alarmingly different processing times. For instance, it is well-known that, in general, neural networks take much more processing time than decision-tree learners [Michie et al. 94].

Let E_{NN} and T_{NN} denote the error rate and the processing time associated with a given neural network algorithm. Similarly, let E_{DT} and T_{DT} denote the error rate and the processing time associated with a given decision-tree learner. Suppose that, for a given database, we have $E_{NN} < E_{DT}$ and $T_{NN} >> T_{DT}$. How can we compare the cost-effectiveness of both algorithms?

The answer is that we can apply the above proposed principle of transformation between efficiency and effectiveness. In this case, the decision-tree learner is more efficient than the neural network. Hence, the saving of processing time associated with the decision tree learner (i.e. T_{NN} - T_{DT}) can be transformed into an effectiveness gain for the decision tree learner itself. One way of implementing this idea is to run the decision-tree learner n times, where $n = T_{NN} / T_{DT}$, with a different setting of parameters each run. (For instance, we could vary pruning parameters, attribute-selection measures, etc.) Then we compare the smallest error rate of all decision-tree learner runs against the error rate of the neural network. Since both kinds of algorithm were given the same time to perform classification, the kind of algorithm with the smallest error rate can be said to be more cost-effective than the other.

Intuitively, this is a fair way of comparing the cost-effectiveness of such different kinds of algorithm as decision-tree learners and neural networks.

This Section discussed how the principle of transformation between efficiency and effectiveness can be applied to the evaluation of cost-effectiveness of algorithms performing the data mining step of the KDD process. In the next Section we show that this general principle is not restricted to the data mining step of the KDD process. It can be applied to pre-processing steps such as attribute selection as well.

4 Applying the Principle of Transformation Between Efficiency and Effectiveness to Attribute Selection Algorithms.

4.1 An Overview of Attribute Selection Methods.

Attribute selection, or feature selection, consists of selecting, out of all attributes potentially available for the KDD algorithm, a subset of attributes relevant for the target KDD task. The selected subset of attributes is then given to a KDD algorithm, here called the target KDD algorithm. One important motivation for attribute selection is to minimize the classification error rate on unseen tuples (the error rate, for short). Indeed, the existence of many irrelevant/redundant attributes in a real-world database may “confuse” the target KDD algorithm, leading to an unduly high error rate [Koller & Sahami 96], [John et al. 94], [Caruana & Freitag 94].

There are two major approaches to attribute selection, namely the wrapper and the filter approaches. In the wrapper approach the training set is divided into two subsets: the training subset and the evaluation subset. Then a heuristic search is done in the space of subsets of attributes. In this search the quality of a subset of attributes is computed in two steps. Firstly, the target algorithm itself is trained on the training subset by using only the subset of attributes being evaluated. Secondly, the error rate of the discovered rules on the evaluation subset is measured and it is directly used as a measure of the quality of the subset of attributes being evaluated.

Attribute-selection methods following the wrapper approach are discussed e.g. in [Aha & Bankert 95], where the space of attribute subsets is searched by a beam search technique; in [Moore & Lee 94], where that space is searched by a schemata search; and in [Bala et al. 95], where that space is searched by a Genetic Algorithm.

In any case, attribute selection methods following the wrapper approach need to apply the target KDD algorithm to the training subset a number of times. Hence, in general the time complexity of the target KDD algorithm is multiplied by the number of applications of the target KDD algorithm. This number can be on the order of thousands or more, in the case of real-world databases with a large number of attributes. This tends to make the wrapper approach very inefficient[2].

In contrast, in the filter approach the quality of a given subset of attributes is evaluated by some method that does *not* use the target KDD algorithm. For instance, Relief [Kira & Rendell 92], [Kononenko 94] uses a randomized technique inspired on the Instance-Based Learning paradigm. This allows it to handle interaction among attributes in a natural way. Relief is quite efficient, having a time complexity roughly linear in both the number of tuples and the number of attributes in the training set.

Another filter algorithm is proposed by [Koller & Sahami 96]. This algorithm takes into account the difference between: (a) the probability distribution of classes given the values of other attributes in the original database; and (b) the corresponding distribution after the removal of an attribute. It performs a backward search which iteratively removes the attribute whose removal minimizes the above difference. Although the time complexity of this algorithm is higher than Relief's one, it is still significantly smaller than the time complexity of most wrapper techniques.

The ADHOC algorithm [Richeldi & Lanzi 96] tries to combine advantages of both wrapper and filter techniques. The meaning of effectiveness in ADHOC is broader than in most attribute selection algorithms (i.e. it goes beyond minimizing error rate, involving a deeper data analysis). However, this algorithm is not very efficient, since its processing time is exponential in the number of attributes.

Overall, wrapper techniques tend to be more effective - i.e. lead to a smaller error rate - than filter ones, since the attribute selection process is carefully tailored for the given target KDD algorithm. However, filter techniques tend to be significantly more efficient - i.e. take a significantly smaller processing time - than wrapper ones.

[2] To improve the efficiency of the wrapper approach, [Caruana & Freitag 94] show that caching results from previous executions of the target algorithm can save a significant amount of time in the execution of the attribute-selection algorithm. However, this caching is useful only for some target KDD algorithms.

4.2 The Trade-Off between Effectiveness and Efficiency in Attribute Selection.

The literature often argues for the superiority of the wrapper model over the filter model [John et al. 94], [Aha & Bankert 95]. However, this argument is based on a single evaluation dimension, namely the effectiveness (as measured by the minimization of the error rate) of the attribute-selection method. However, as mentioned before, the wrapper approach tends to be significantly more computationally expensive (i.e. less efficient) than the filter approach.

In this Section we discuss our approach to take into account the trade-off between effectiveness and efficiency in attribute selection, by focusing on a combined cost-effectiveness measure of attribute selection methods. Let E denote the error rate achieved by applying the target KDD algorithm to the reduced database containing only the selected attributes, and let T denote the processing time taken by the attribute selection technique. Furthermore, let E_W and T_W denote the error rate and the processing time associated with a given wrapper method W. Analogously, let E_F and T_F denote the error rate and processing time associated with a given filter method F.

Following the spirit of this paper explained in the Introduction, we are particularly interested in the commonplace situation where $E_W < E_F$ but $T_W >> T_F$, i.e. the wrapper method W leads to an error rate smaller than the filter method F, but the method W takes much more processing time than the method F. The arguments presented below can be easily generalized for situations involving more than two attribute selection methods, regardless of they belong to the wrapper or filter approach.

In order to evaluate the cost-effectiveness of attribute selection methods, we apply the general principle proposed in Section 2, that is: *a gain in efficiency can be transformed into a gain in effectiveness*. In the context of attribute selection, the saving in processing time achieved with the filter approach can be transformed into an additional reduction of error rate.

More specifically, using the previously defined notation, the saving in processing time due to the use of the filter approach is given by $T_W - T_F$. We can then spend this saved time by applying several KDD algorithms other than the target one to the reduced database produced by the filter approach. The error rate associated with this kind of filter approach will then be the minimum error rate among the error rates achieved by all KDD algorithms applied to the reduced database (including the target algorithm). Since the idea of applying several KDD algorithms to a database and selecting the best result is usually called a toolbox approach in the literature, we call the above described approach a filter/toolbox approach.

Note that this combined filter/toolbox approach is possible due to the flexibility of the filter approach, which selects attributes in a way independent of the target KDD algorithm. In contrast, a combined wrapper/toolbox approach would not make sense, since in the wrapper approach the attribute selection method is tailored for a single target KDD algorithm.

Let E_{FT} denote the minimum error rate associated with the filter/toolbox approach. Now, since both the filter/toolbox and the wrapper approach were allowed to use the same amount of time to reduce the error rate, we have a fair criterion to compare the *cost-effectiveness* of these approaches. More specifically:

if $E_{FT} < E_W$, then the combined filter/toolbox approach is more cost-effective than the wrapper one;
if $E_{FT} > E_W$, then the wrapper approach is more cost-effective than the combined filter/toolbox one.

Obviously, the values of E_{FT} and E_W depend on both the database and on the target KDD algorithms applied to it. Hence, in order to state that a given approach (wrapper or filter/toolbox) is in general more cost-effective than the other, we would need to perform a large number of experiments, with many databases and target KDD algorithms. Such experimental work is left for future research. Here we limit ourselves to say that, intuitively, we believe that the filter/toolbox approach is often more cost-effective than the wrapper one. This belief is backed by the following two complementary arguments.

Firstly, as discussed before, wrapper techniques tend to be much more time-consuming than filter ones. Hence, this large time saving associated with the filter approach is likely to allow the application of several KDD algorithms to the reduced database. This is also facilitated by the fact that the KDD algorithms are always applied to the final reduced database, which is expected to have a small subset of selected attributes. Note that this is an advantage over the wrapper approach, where the target KDD algorithm sometimes has to be applied to a relatively large subset of selected attributes.

Secondly, it has been shown, both theoretically and empirically, that the error rate of a KDD algorithm is strongly dependent on the database [Schaffer 94], [Michie et al. 94], [Kohavi et al. 96]. The difference in the error rates of two KDD algorithms can be very significant, and it is very difficult to decide *a priori*, in the wrapper approach, which target KDD algorithm should be applied to a given database to minimize error rate. Hence, the combined filter/toolbox approach can lead to a large reduction of error rate that often cannot be achieved with wrapper approach.

5 Conclusions.

Most of the KDD literature views effectiveness and efficiency as very different dimensions along which a KDD technique can be evaluated. Hence, when the performance of a KDD technique along both these dimensions is analyzed, this analysis is done separately for each dimension. This is in essence a reductionist approach. In contrast, this paper proposes a more holistic approach, where effectiveness and effectiveness are taken into account together to derive a unifying measure of cost-effectiveness of KDD techniques. (In passing we remark that, in general, exact sciences are becoming more and more holistic[3].)

Our approach to evaluate the cost-effectiveness of KDD techniques is based on the principle that a gain in efficiency can be transformed into a gain in effectiveness. To show the generality of this principle and to show how it can be actually used to derive a fair measure of cost-effectiveness, we discussed the application of this principle in two very general cases. Firstly, we discussed its application to the data mining step of

[3] The typical example is physics - see e.g. [Davies 83] for an excellent discussion about holism versus reductionism in the modern physics.

the KDD process (in the context of classification). Secondly, we discussed its application to attribute selection, a major pre-processing step of the KDD process.

In particular, the latter case of application of our principle was discussed in more detail, since we took the opportunity to clarify a "misconception" - or at least an unduly narrow view - commonly found in the attribute selection literature. This literature often emphasizes that wrapper methods are "superior" to filter ones because the former are more effective. However, we argued that, using our holistic approach to evaluate the cost-effectiveness of attribute selection methods, it is the other way round. Filter methods tend to be more cost-effective than wrapper ones. However, we remark that our arguments concerning this point still need to be validated by empirical evidence, which is left for future research.

References.

[Aha & Bankert 95] D.W. Aha and R.L. Bankert. A comparative evaluation of sequential feature selection algorithms. *Proc. 5th Int. Workshop on Artif. Intel. and Statistics*, 1-7. Ft. Lauderdale, FL. 1995.

[Bala et al. 95] J. Bala, J. Huang, H. Vafaie, K. DeJong and H. Wechsler. Hybrid learning using genetic algorithms and decision trees for pattern classification. *Proc. 14th Int. Joint Conf. AI (IJCAI-95)*, 719-724. 1995.

[Caruana & Freitag 94] R. Caruana and D. Freitag. Greedy attribute selection. *Proc. 11th Int. Conf. Machine Learning*, 28-36. 1994.

[Davies 83] P. Davies. *God and the New Physics.* Penguim Books, 1983.

[John et al. 94] G.H. John, R. Kohavi and K. Pfleger. Irrelevant features and the subset selection problem. *Proc. 11th Int. Conf. Machine Learning*, 121-129. 1994.

[Kira & Rendell 92] K. Kira and L.A. Rendell. The feature selection problem: traditional techniques and a new algorithm. *Proc. 10th Nat. Conf. AAAI*, 129-134. 1992.

[Kohavi et al. 96] R. Kohavi, D. Sommerfield and J. Dougherty. Data mining using MLC++: a machine learning library in C++. *Technical Report*, Silicon Graphics Inc., 1996.

[Koller & Sahami 96] D. Koller and M. Sahami. Toward optimal feature selection. *Proc. 13th Int. Conf. Machine Learning*, 1996.

[Kononenko 94] I. Kononenko. Estimating attributes: analysis and extensions of RELIEF. *Proc. 1994 European Conf. Machine Learning, LNAI* 784, 171-182. 1994.

[Michie et al. 94] D. Michie, D.J. Spiegelhalter and C.C. Taylor. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, 1994.

[Moore & Lee 94] A.W. Moore and M.S. Lee. Efficient algorithms for minimizing cross validation error. *Proc. 11th Int. Conf. Machine Learning*, 190-198. 1994.

[Quinlan 94] J.R. Quinlan. The Minimum Description Length Principle and categorical theories. *Proc. 11th Int. Conf. Machine Learning*, 233-241. 1994.

[Quinlan & Rivest 89] J.R. Quinlan and R.L. Rivest. Inferring decision trees using the minimum description length principle. *Info. and Computation* 80(3), 1989, 227-248.

[Quinlan & Cameron-Jones 95] J.R. Quinlan and R.M. Cameron-Jones. Oversearching and layered search in empirical learning. *Proc. 14th Int. Joint Conf. AI (IJCAI-95)*, 1019-1024.

[Richeldi & Lanzi 96] M. Richeldi and P.L. Lanzi. Performing effective feature selection by investigating the deep structure of data. *Proc. 2nd Int. Conf. Knowledge Discovery & Data Mining*, 379-382. 1996.

[Schaffer 94] C. Schaffer. A conservation law for generalization performance. *Proc. 11th Int. Conf. Machine Learning*, 259-265. 1994.