Attribute Selection by Measuring Information on Reference Distributions

Rómer Rosales and Olivier Chapelle
Yahoo! Advertising Sciences and Research
{romerr,chap}@yahoo-inc.com

Abstract

A great number of services, experiments, and decisions at Yahoo! require analyzing rich data sources. This data almost invariably holds a large number of attributes. In these scenarios, the efficient selection of relevant attributes is imperative for data analysis (*e.g.,* modeling, prediction). When approaching new data analysis tasks, domain experts, researchers, and engineers spent a considerable amount of resources identifying (manually or semi-automatically) these relevant attributes. This paper attempts to address this problem by providing a simple and largely automated attribute selection approach. The method is based on reformulating the mutual information (MI) measure. We show why MI cannot in general be used effectively without considerable domain expertise and describe a more appropriate measure that allows for a much larger level of automation (removing considerable manual work from the analysis loop). Experiments on the tasks of predicting clicks and conversions for Yahoo! display advertising platform in the context of the NGDStone project show the effectiveness of the proposed approach.

1 Introduction

This paper addresses the problem of automated selection of relevant data attributes (also referred to as co-variates or features) when only some attributes can be utilized for prediction/data modeling tasks. This is a very familiar scenario when working with rich data sources common at Yahoo!, where the available number of possible attributes is too large to be used in their entirety during prediction/modeling. Practical considerations, such as memory, latency, and training time constraints, make attribute selection a clear requirement in many real tasks. Additionally, non-informative attributes can introduce noise and reduce the predictive accuracy of the system.

While expert knowledge should be used when available, this often serves as a rough guide as the number of attributes is too large to be considered for a systematic and thorough solution. Normally the available knowledge is limited or expensive to incorporate in practice and thus automated methods are essential.

This paper focuses on the case when attributes and target value(s) are discrete. Optimal feature selection is known to be an NP-hard problem in general [4], where the complexity grows at least exponentially with the number of attributes considered. In machine learning, most methods for approaching the selection problem are broadly classified into filter and wrapper methods [2]. The first attempts to identify the optimal attributes independently of the machine learning model while the latter involves incorporating model-specific learning in the selection.

We focus on filter methods as they are more practical in problems with a large number of data attributes. The classical and widely employed method in this category is the selection of attributes based on each attribute's information content about the target attribute [6], where the Mutual Information [5, 7] is employed to measure the probabilistic dependence between discrete random variables. This is referred to, in this report, as the Standard Mutual Information (SMI) method, and consists of sorting the attributes' importance based on their MI score. This method has some limitations, including its reliance on empirical probability estimates to compute the MI. One way to address this was studied in [3, 8] by considering (approximate) distributions over the MI value (rather than a point estimate). When building predictive models, where the central motivation is in *learning* a model with training data and predicting the target variable on unseen test data, the differences between the empirical training and test distributions can lead to meaningless estimates of MI. This leads to inappropriate attribute selection results.

In the following presentation we focus on this class of problems, illustrate them in the context of Yahoo!'s data, and propose a reformulation to address them. This reformulation is based on utilizing a reference distribution (separate from the training distribution) to calculate the MI score. We will show theoretically and experimentally that this development allows us to select attributes that are more suitable for the problem at hand. In particular we argue how the proposed approach can considerably reduce a) the time needed for manual attribute testing and selection and b) the amount of expert knowledge required to build appropriate data models for prediction tasks.

2 Formulation

Let the available training data be represented by a set of N tuples (data points) $\mathcal{D} = \{(x_1, \ldots, x_D, y)\}_N$, where x_i represent the i-th data attribute/feature and y represents the target value for the tuple. We let the each tuple be a sample from an unknown distribution p over random variables $X = (X_1, ..., X_D)$ and Y.

Standard Mutual Information Criterion The definition of mutual information (MI) between a random variable X_i (normally representing an attribute) and a target random variable Y is given by:

$$I(X_i, Y) = \sum_{x_i, y} p(x_i, y) \log \frac{p(x_i, y)}{p(x_i)p(y)} \tag{1}$$

where $p(x_i, y)$ represents the joint probability of $X_i = x_i$ and $Y = y$, and $p(x_i)$, $p(y)$ are the corresponding marginals. For a $\log$ function with base 2, $I(X_i, Y)$ indicates how many bits knowing about (the value of) X_i tells us about (the value of) Y. A useful property for the MI is that $I(X_i, Y) \leq H(Y)$[1], the information provided about a discrete random variable cannot be larger than its entropy.

The MI score offers a clear guide for attribute selection. The standard procedure consists of sorting all the attributes of interest according to their information content about the target variable Y and selecting the top K. The MI for a conjunction of attributes X_C can be computed by letting X_C take values on the product space of the attributes in question (thus, basically representing a new compound attribute). This combination of attributes in general increases the space and time complexity exponentially with the number of attributes. Unless combinations of all possible attributes are considered, this procedure is not guaranteed to provide the global optimal solution.

Here we concentrate on a key drawback of this method. This can be illustrated as follows: Let X_u be a random variable taking unique values (X_u can for instance be an event identifier), then $I(X_u, Y) = H(Y)$ since the values of X_u can fully identify the data point and therefore its label. Formally:

$$\begin{aligned} I(X_u, Y) &= \sum_{x,y} p(x_u, y) \log p(y|x_u)/p(y) \quad (2) \\ &= \sum_{y} p(y) \log 1/p(y) = H(Y), \quad (3) \end{aligned}$$

since $p(y^*|x_u) = 1$ for some $y = y^*$ (zero otherwise). However, X_u is useless as an attribute for predicting y since its values are unique and not observed in any test set.

Although the correspondence needs not be one-to-one, the above case highlights the main concern that the values taken by X_u are identifying the data points, and are not necessarily of use for predicting the target variable on different data. More generally, the larger the number of different values an attribute can take, the higher its mutual information could potentially be, but also the higher the risk that it does not *generalize* on the test distribution.

Reformulating the Mutual Information Criterion In selecting relevant attributes, we are mainly concerned with performance on a test or reference set. Ideally this test set follows a distribution similar to the available training set. However when this is not the case, like in the cases highlighted above, the MI is not a valid relevance score.

In order to address this problem we propose using a related function that explicitly considers a reference distribution. Let the reference distribution be given by $\tilde{p}(x, y)$, then define the MI with respect to the reference distribution by:

$$I_{\tilde{p}}(X_i, Y) = \sum_{x_i, y} \tilde{p}(x_i, y) \log \frac{p(x_i, y)}{p(x_i)p(y)}, \tag{4}$$

where the difference lies on calculating the expectation with respect to the reference (not training) distribution.

This definition has the problem that the $\log$ ratio is undefined for cases when $p(x_i) = 0$. This happens when an attribute value has been seen in the reference distribution $\tilde{p}$ but not in the training set distribution p. Thus, we utilize a smoothing of the training data distribution of the form:

$$p_r(x_i, y) = \frac{Np(x_i, y) + p(y)}{N + |X_i|}, \tag{5}$$

guaranteeing $p_r(x_i) > 0$, where $|X_i|$ is the number of states taken by X_i. It is possible to show that, if $(\forall y)p(y) > 0$, this does not affect the target distribution, that is: $p_r(y) = p(y)$. In the critical case where X_j does not appear in the training data distribution, we can show that $p_r(y|x_i) = p(y)$. In the latter case we have that the log ratio above becomes 0.

The main relevant property of the new information quantity is that as attributes are evaluated on a reference distribution, spurious relationships (such as those seen above) found in a specific dataset and that do not generalize to the test dataset are mostly ignored. Consider an extreme example where the values of X_u seen in the training data do not appear in the test distribution. Using the new definition, the test distribution will place no mass to these values, and X_u will have no measured information about the target variable of interest. More formally:

$$I_{\tilde{p}}(X_u, Y) = \sum_{x_u, y} \tilde{p}(x_u, y) \log \frac{p_r(x_u, y)}{p_r(x_u)p_r(y)} = 0. \tag{6}$$

We note that in many instances, any reference distribution $\tilde{p}$ computed on a valid sample (not necessarily the test data)

[1] $H(Y) = -\sum_y p(y) \log p(y)$ denotes the entropy[1] of Y.

Table 1: Top features for click prediction along with their mutual information. First table: standard mutual information; second and third table: modified mutual information. Third table contains the top conjunction features.

Single feature	SMI (bits)
event_id	0.59742
query_string	0.59479
xcookie	0.49983
user_id	0.49842
user_segments	0.43032
Single feature	**RMI (bits)**
section_id	0.20747
served_creative_id	0.20645
site	0.19835
served_campaign_id	0.19142
rmx_ad_grp_id	0.19094
Conjunction feature	**RMI (bits)**
section_id x served_advertiser_id	0.24691
section_id x served_creative_id	0.24317
section_id x served_IO_id	0.24307
served_creative_id x publisher_id	0.24250
served_creative_id x site	0.24246
site x served_advertiser_id	0.24234
section_id x pixeloffers	0.24172
site x served_IO_id	0.23953
publisher_id x served_advertiser_id	0.23903

Table 2: Top features for conversion prediction along with their mutual information. First table: standard mutual information; second and third table: modified mutual information. Third table contains the top conjunction features.

Single feature	SMI (bits)
event_id	0.03102
receive_time	0.03059
query_string	0.02963
xcookie	0.02925
user_id	0.02923
Single feature	**RMI (bits)**
conversion_id	0.02338
served_IO_id	0.02207
rmx_ad_grp_id	0.02136
served_campaign_id	0.02090
served_advertiser_id	0.02082
Conjunction feature	**RMI (bits)**
conversion_id x offer_type_id	0.02379
conversion_id x pop_type_id	0.02369
conversion_id x served_IO_id	0.02347
served_IO_id x offer_type_id	0.02267
served_IO_id x served_bid_type	0.02235
served_advertiser_id x pop_type_id	0.02120
served_advertiser_id x offer_type_id	0.02108
advertiser_network_id x offer_type_id	0.00993
publisher_network_id x advertiser_network_id	0.00770

different than the training distribution will allow the proposed measure to avoid spurious relationships particular to the training data. However, a reference distribution closer to the test distribution is preferred as test-specific dependencies will be better captured by the new MI definition.

3 Experiments

In order to test the proposed approach, we focused on two central prediction tasks in the context of Yahoo!'s display advertising platform: 1) predicting a user ad click given a page serve and user context, and 2) predicting a conversion/action given that a click has occurred in this context. Both of these tasks are at the core of performance-based advertising products, as they are required to calculate expected price, revenue/profit measures, and user satisfaction metrics.

The data utilized for our experiments consisted of a sample of Non-Guaranteed Display (NGD) logs (serve, click, and conversion events). For the problem of attribute selection for click prediction, we considered one day of data for the training distribution and one day for the reference distribution. Since there are many more serves than clicks, the serves were further sub-sampled. After filtering (spam removal, etc.) and joining the appropriate logs, the resulting data set had the following statistics: about 78M (million) events with a CTR of 28% in both the training and reference distributions. For conversion prediction, the data consisted of logs for a period of 5 days (training), 1 day (reference), and 2 days (testing). The statistics of this data set were: 125M events for training, 25M for reference, and 50M for testing.

Our goal is to identify predictive features in the most automated manner possible (reducing time spent by people on this task). Thus, practically all the data attributes provided by the RMX logs are considered potential features. These were considered in its original form, without feature pre-processing. They include identifiers for the actual (serve/click/conversion) event, advertiser, publisher, campaign, bcookies, timestamps, advertiser/publisher specific attributes, related urls, demographics, user-specific attributes (*e.g.,* assigned segments), etc. We consider conjunctions of any of these attributes, giving rise to thousands of possible compound features in practice. Each feature in turn can take from two to millions of possible values.

The important element to consider is that without time-consuming research into attribute definitions, it is extremely tedious to apply most machine learning or data mining/analysis algorithms for. Thus, requiring considerable effort from *e.g.,* machine learning scientists or domain experts. Wrapper methods are not appropriate in this setting as they require training using a large set of variables; this is usually impractical except for some simple models. It is in this setting where filter methods, such as the MI methods described here, can be more advantageous.

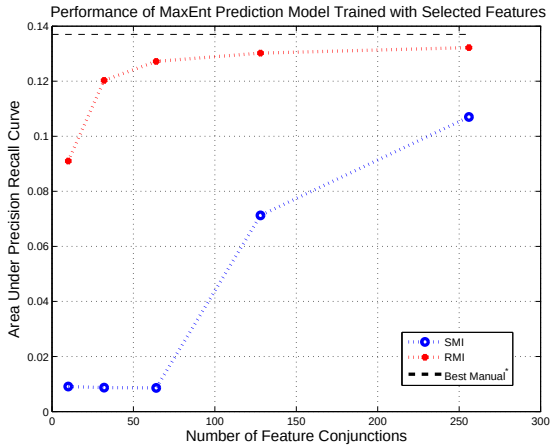

Figure 1: Performance of Logistic Regression Models trained with the top K features given by SMI and RMI

3.1 Attribute Selection Results

We applied the standard MI (SMI) ranking algorithm for feature selection. The results, summarized in Tables 1-2(top) reflect our main concern. In the presence of spurious features, or features that are informative about the data point *per se* rank substantially high. The calculated MI score is correct in that it reflects the information content of these features; however, these features are too specific to the training data distribution.

The proposed extension of the MI score utilizing a reference distribution (RMI) provides a more appropriate ranking as shown in Tables 1-2(mid-bottom). The reason for this is that the information content is calculated with respect to (expectations on) the reference distribution and thus feature values that are not seen in the new distribution are basically considered less important and their impact on the information score is reduced.

More specifically, attributes such as `event_id` that identifies the data point have maximal information content according to the training distribution (SMI), but near zero information content when calculated with a reference distribution (RMI) (*c.f.*; Sec. 2). A similar effect was observed for other features that have low relevance for prediction such as `query_string` and `receive_time` which unless parsed are too specific, `xcookie` and `user_id` which clearly do not generalize across users (but could be quite informative about a small fraction of the test data), and `user_segments` which is encoded as a string with a list of segments. The results for other features are more subtle but follow the same underlying principle where a reference distribution is utilized to avoid spurious dependencies often found when utilizing empirical distributions.

3.2 Learning Performance Results

We now explore the question of whether the new feature rankings actually offer any performance gains. For this, we ranked all the feature conjunctions provided by SMI/RMI and trained a Logistic Regression model using the top K conjunctions given by each method. The results for conversion prediction are shown in Fig. 1 for various K values.

The graph shows performance in terms of the (scaled) Area Under the Precision-Recall curve (similar results obtained when measuring the Area Under the ROC curve and percentage of correct predictions). The results indicate that SMI ranks very irrelevant features on top. After more than 100 conjunctions, more relevant features start to be included in the model as evidenced by the performance gains. On the contrary, RMI captures relevant features much earlier, as only the top 10 features are sufficient to perform better than SMI with 128 features. The performance increases considerably at 32 features, and appears to stabilize after 64 features. One interesting question is whether after a large K the performance of both methods will be comparable. As seen in both curves, when 256 features are utilized, still the difference is considerable. We believe the performance of SMI will remain affected by the various irrelevant feature included early on, depending on how much the model is susceptible to the noise introduced by these.

Finally, the top line in the graph represents the current best model after considerable feature engineering (*e.g.,* where features such as `receive_time` and `query_string` have been parsed and transformed into a suitable representation), manual feature selection, grouping, and train/test exploration experiments. This indicates that the proposed selection method produced quantitatively comparable results but have been much more resource-efficient than the best possible modeling efforts in this problem so far.

4 Conclusions

We have developed and tested an efficient filter approach for attribute selection. The approach is based on reformulating the widely used mutual information measure to address some of its limitations for attribute selection. These limitations are of conceptual and practical nature particularly in cases where training and test data distributions are inherently different for some attributes or cannot be estimated accurately due to limited data. This is very often the case when working in real application scenarios with a large number of attributes. We have found that this approach allows for a considerable increase in automation as it can efficiently assign a lower rank to spurious features as seen in our experimental evaluation. We expect this to be a valuable tool for approaching new problems or analysis tasks as it allows for a quicker and less resource-intensive initial understanding of new data sources.

References

[1] T. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley Interscience, New York, NY, USA, 1991.

[2] I. Guyon and A. Elisseeff, editors. *JMLR Special Issue on Variable and Feature Selection*. Journal of Machine Learning Research, 2003.

[3] M. Hutter. Distribution of mutual information. In *Advances in Neural Information Processing Systems 14*, pages 399–406, Cambridge, MA, 2002. MIT Press.

[4] L. Hyafil and R. L. Rivest. Constructing optimal binary decision trees is np-complete. *Information Processing Letters*, 5(1):15–17, 1976.

[5] S. Kullback. *Information Theory and Statistics*. Dover, 1968.

[6] D. Lewis. Feature selection and feature extraction for text categorization. In *Speech and Natural Language Workshop*, pages 212–217, 1992.

[7] D. Lindley. On a measure of the information provided by an experiment. *Annals Math. Stat.*, 27:986–1005, 1956.

[8] M. Zaffalon and M. Hutter. Robust feature selection by mutual information distributions. In *Proc. 18th International Conf. on Uncertainty in Artificial Intelligence (UAI-2002)*, pages 577–584. Morgan Kaufmann, San Francisco, CA, 2002.