Feature Selection: We've barely scratched the surface

George Forman
Intelligent Enterprise Technologies Laboratory
HP Laboratories Palo Alto

HPL-2005-165
September 19, 2005*

trends, future work, feature selection, machine learning, classification, document categorization

* Internal Accession Date Only
Published in IEEE Intelligent Systems, November 2005

Feature Selection: We've barely scratched the surface

An essay requested for IEEE Intelligent Systems, Trends and Controversies

George Forman, Hewlett-Packard Labs

September 9, 2005

Selecting which inputs to feed into a learning algorithm is important and yet often underappreciated. People usually talk about 'the' clusters in a dataset. But if you were to cluster the vehicles in a parking lot into groups, your answer would depend completely on which features you considered: color? model? license plate? Without prior knowledge of which sorts of clusters are desired, there is no right or wrong choice. However, if you were paid to generate a *predictive* model for gas mileage, you would consider vehicle weight and ignore color. These examples are meant to be obvious, but real-world datasets tend to involve large and often complex feature selection choices, whether or not they are made deliberately.

If feature selection is done poorly, no clever learning algorithm can compensate, e.g. predicting gas mileage from color and trim. If done well, the computational and memory demands of both the inducer and the predictor can be reduced, and usually more importantly, the prediction accuracy improved: the performance of Naïve Bayes—ever popular for its ease of programming—is highly sensitive to feature selection; even relatively insensitive algorithms, such as Support Vector Machines, can benefit substantially. In some circumstances, such as biochemistry wet labs, eliminating all but essential features can reduce the cost of obtaining measurements. Finally, feature selection by itself has useful applications, such as the "Statistically Improbable Phrases" now appearing at Amazon.com to help end-users characterize books.

While several good feature selection techniques are known, I contend that feature selection is still in its infancy and major opportunities await. In this essay I will highlight several key prospects, as I see them. For readers wishing a survey, a tutorial or advanced articles on feature selection, I refer them to the 2003 special issue on variable and feature selection in the online Journal of Machine Learning Research [2] or to the recent survey by Liu and Yu [3].

Low Hanging Fruit

A first avenue is simply to bring known successful techniques into mainstream usage. Too often an available dataset is used 'as-is' with all its features, however they came to be. Much more thought is generally given to the induction algorithms. Part of the solution lies in just streamlining user interfaces to make automated feature selection part of the natural process.

Of course, people don't want to be bothered with more knobs to tune. Just as cross-validation can be used to select which of several learning models performs best for a given training set, so too can it extend to automating feature selection decisions.[1] But this has its limits: cross-validation on large datasets can exceed the user's patience budget, and cross-validation on small training sets is more likely to produce overfit models than true improvements in generalization accuracy. This can be combated with prior knowledge about which combinations of feature selection and learning algorithms perform well for

[1] Cross-validation involves breaking a dataset into, say, 10 pieces, and on each piece, testing the performance of a predictor build from the remaining 90% of the data. In this way, one can estimate how well each of several learning algorithms perform on the available data. The best is then chosen to learn on all of the data.

different kinds of data. This is an open opportunity for meta-learning research.

Accuracy vs. Robustness

While a great deal of machine learning research seeks ever to improve accuracy, it sometimes comes at a cost in brittleness. To enable more widespread use of feature selection, there is a valuable vein of research in developing robust techniques. We at Hewlett-Packard have faced industrial datasets where most feature selection techniques fail spectacularly. For example, in a multi-class task for document classification where one class is very easy to predict, e.g. German documents, most feature selection methods will focus on the many strongly predictive foreign word features for the easy class, leaving the other classes hard to distinguish [1]. Although we devised a solution for this specific type of problem, certainly more research into robust methods is called for. I urge practitioners to share the failures they encounter on real datasets; most of the public benchmark datasets do not expose these issues.

Trends

I predict several trends will increase the demand for feature selection. One is obviously the growing size of datasets, requiring either random sub-sampling of rows or purposeful feature selection of columns. The former is easier, but the latter may be more beneficial, and may be the only choice for 'wide' datasets with many more columns than rows (e.g. >100,000 features in genomics or document classification).

And datasets are generally widening with the increasing ability to link to additional databases and join with other tables. In the running example, each vehicle could be linked to external databases with pollution ratings, sales figures, and/or review articles, potentially adding many thousands of features. Today such linking requires human thought and effort, but tomorrow such linking may be automated. This increases the pressure on automated feature selection to efficiently determine *useful* widening. The demand for this research will come primarily from practitioners who seek optimal prediction for economically valuable tasks, not from pure machine learning researchers who care about optimizing performance on fixed, self-contained benchmark datasets for comparable, publishable results.

Rich Data Types

The trend toward richer data types is pushing feature selection in both scale and complexity. Natural language text features and image features are becoming commonplace, e.g. the relatively mature area of document classification. To handle rich data types, a *feature generator* replaces them by many features of primitive data types. For example, in the *bag of words* model for text, each unique word in the collective corpus generates a unique feature, e.g. an integer representing the number of times that word occurred in each record. The number of generated features can become quite large if the vocabulary present in the corpus is large, e.g. long document texts, especially when multiple natural languages are present.[2] Or there may be multiple text fields to be expanded into separate sets of features, since the word "Smith" appearing in the title should be treated differently than if it appears in the author field.

Considering the rich, expressive power of human language to address any topic, a simple bag of words model gives an extremely limited view—the relative positions of the words are lost (reading sorted the Try with words). By adding a feature for each two- and three-word phrase that appears in the corpus, the bag-of-phrases representation can disambiguate a "light car" from a "car light," at the cost of many more potential features to consider. Other feature generation techniques may link words and phrases to external databases with additional information to generate even more features,

[2] The widespread text processing techniques of lowercasing, eliminating common stopwords, and stemming words to their root form reduce the total number of features by only a small amount.

such as thesauri and controlled-vocabulary taxonomies. With the deluge of hierarchically nested XML data types and time-based multi-media, feature generation research will continue to expand the possibilities.

In all, the potential space for feature generation from rich data types is enormous and not all worthwhile. Rather than attack it simply in terms of greater scale, there will be a need to integrate feature selection with the feature generation process, just as conventional breadth-first and A* search techniques carefully coordinate state generation with evaluation. After all, inducing predictive models can be stated as a search problem that considers variations in feature generation, feature selection, induction algorithms and their associated parameters. While this may sound quite involved today, CPU cycles will increasingly be cheaper than an expert's time.

That said, we can quickly fall into the trap of overfitting our data if our search space is large and the training set relatively small. For example, given only a few training examples, it may happen that color can help predict gas mileage in cross-validation, but we wouldn't expect this correlation to generalize to larger datasets. Once again, meta-learning methods are called for to help guide the search in the absence of large amounts of training data for a particular new problem. Likewise, machine-readable domain knowledge could help constrain the search to meaningful correlations. We don't know how to automate this today, but hopefully we will one day.

Cost & Time

As if the present challenges weren't already enough, real-world problems not uncommonly include a cost and time delay for obtaining (additional) features. For example, Veeramachaneni, et al. [4] describe a practical biomedical problem where additional medical tests may provide predictive features but at a cost. They then go on to develop an elegant algorithm to maximize predictive performance in a cost-efficient manner. This is in contrast to typical *active learning* problems where the cost is entirely in obtaining class labels.

Time plays an additional role in some non-stationary domains where the best features have a seasonal dependency. For example, in spam filtering, the word "Christmas" is useful in December, but then has fairly low value for the following months. There are many similar time component issues associated with click stream mining of web sites and shopping behavior.

I cannot claim that cost and time represent current trends in research, but I foresee their need in practical deployment and expect these areas will eventually see greater activity.

A Vision

One of the reasons why C4.5 decision trees are popular is that they can handle a heterogeneous collection of features types (mixed nominal, integer, and real) without requiring any special consideration by the user. Although I stated earlier that too little attention is often paid to feature selection, in my vision of future machine learning platforms, no special consideration by the user will be needed. Instead, a robust feature selection subsystem equipped with meta-knowledge will seamlessly handle heterogeneous types, linked database widening, etc. Getting there will require much stimulating research, fueled by real-world problems brought to light by practitioners. Any takers?

References

1. Forman, G. A pitfall and solution in multi-class feature selection for text classification. In *Proceedings of the Twenty-First international Conference on Machine Learning* (Banff, Canada, July 04 - 08, 2004). ICML '04, vol. 69, 2004.
2. Guyon, I. and Elisseeff, A., eds. Special Issue on Variable and Feature Selection. *Journal of Machine Learning Research*, vol. 3 (Mar), 2003. (introduction & 15 articles)

http://jmlr.csail.mit.edu/papers/special/feature03.html

3. Liu, H. and Yu, L. Toward Integrating Feature Selection Algorithms for Classification and Clustering. *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491-502, April 2005.
4. Veeramachaneni, S., Demichelis, F., Olivetti, E. and Avesani, P. Active Sampling for Knowledge Discovery from Biomedical Data. *Forthcoming in the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, (Porto, Portugal), PKDD'05, October 2005.