Combining Classifiers to Improve the Error Rate in Numerical Character Recognition

Y2841914

January 26, 2011

Pattern Recognition has been a major area of research since the first computers were developed. The ability to classify a given sample into the correct group is something that humans can do innately, but is relatively difficult for computers to do reliably [1]. Over the last 40 years or so, the subject has progressed in leaps and bounds in terms of the theory of pattern recognition, and has been helped along in the practical aspect by the Moores-Law-esque growth in computational power, making even complex algorithms usable in real-world applications [2].

One of the main problems of pattern recognition is the barrier between the real world and the theoretical world we design inside the computer. For example, computer systems use fonts to represent letters and numbers and can quite easily calculate the value of a single character in such a form. Humans, however, use a very broad range of differing characters to represent the same thing, such as those who cross their sevens or loop their Fs. The human brain is wired up entirely around the concept of pattern recognition, in such a way that abstracting away these differences to uncover the semantic meaning takes very little effort at all [3]. It is this idea that has led to the development of neural networks and bio-inspired classifiers. The focus of this review is on how classifiers are combined to improve the overall classification of a pattern recognition system.

Although the classifiers developed and presented in much of the literature [4],[5],[6] have good classification rates and are often quite reasonable for real-world applications, there are certain situations that require much more confidence about the classification. This idea led onto the introduction of reject classes [7] to signify the classifiers doubt about its result. Traditionally, these rejected patterns would be either completely rejected or passed out to a user to identify separately (using the computer-based system as a method of speeding up the main brunt of the work). In [8], *Kittler* et al. explain that the misclassified/rejected patterns would not necessarily overlap and imply that a complementary classifier could be used to assist the original classifier with those it had rejected. There are many ways of combining multiple classifiers to help with the reduction of overall rejection and error rates and the papers reviewed herein present only a minority of potential combination methods, and focus on the recognition of numerical characters.

In [9], *Cao* et al. suggest that the combination of multiple feature-extraction methods can be used to reduce the rejection rate of patterns by creating a separate feature space for each stage of a multistage classifier. The authors describe two methods of extracting features from one set of data, the first being a down-sample greyscale representation of the pattern, the second being a fuzzyfied version of the directional histogram features developed by *Kimura* et al. [10].

The classifier that [9] implements is based around a neural network architecture, where the inputs to the network are the down-sampled greyscale features from the first extraction method. These features are then incrementally clustered using an algorithm that works in a similar fashion to the ISODATA algorithm [11], splitting and merging clusters until each cluster has a unique class. The output of this stage of the neural network is a pair of numerals that the pattern is closest to in the clustered feature space. This is then passed on to the next stage, unless the distance between the pattern and closest clusters do not meet certain thresholds, in which case the pattern is rejected at this stage.

The idea is to pass the pattern to a subclassifier that deals with these two numerals and classifies the pattern into one of the two classes, or the reject class.

The sub-classifier network is the next stage in the multistage classifier, and consists of a series of 45 sub-classifiers, each corresponding to two classes out of the ten possible. The proposal also includes a Rejection Neuron, trained on the eight remaining classes, to signify that the pattern has been passed to the wrong sub-classifier.

The output of the initial clustering stage is used to decide which of the sub-classifiers should be used in this stage of the classifier (reducing the computational load, as classifying between two classes and a reject is much easier than between ten classes).

In [12], *Huang and Suen* have taken a different approach to the problem of combining multiple classifiers to improve the error rate. They build on their work in [13] developing a method of combining classifiers regardless of the algorithm implemented. This is in stark contrast to [9] where a specific system that took the output of one classifier to influence the activity of the second was developed. In [12], the authors describe a method called the Behaviour-Knowledge Space Method, or BKS-Method. The idea of this method is to allow multiple classifiers to present their outputs to a common output space, where each classifier gets its own dimension to store its findings. This common output space is the BKS, allowing multiple classifiers to simultaneously record their results in a space that can subsequently be used for analysis.

A key aspect emphasised in [12] is the lack of an independence assumption. The independence assumption is present in most other approaches to the classifier combination problem. This implies that for all classifiers used in the combination method, the classifiers are entirely independent of each other. Through the use of the BKS-Method, this assumption is not necessary, as the results can all be stored in the BKS simultaneously.

Both [9] and [12] present classifiers that pass the outputs from one stage to another. [9] uses the output of the input stage to control the flow of execution in the second stage, by disabling most sub-classifiers and enabling just one for the remainder of the classification, whereas [12] takes the output of the first stage and passes it to a space where classification could occur based on the output of the initial classifiers, essentially using the initial classifications as the input vectors to the next classifier.

In [14], *Mitrakis* et al. propose a slightly different method of combining classifiers, although not designed specifically for numerical character recognition. The authors present a system that classifies the inputted data, and if a definitive result is found, the classifier outputs it. If the reject class is outputted instead, the pattern is passed to a more complex fuzzy classifier, designed to handle the ambiguity of the input patterns. This method combines the best part of two complementary classifiers, in that it allows simple, crisp classifiers to efficiently classify the pattern if they recognise it, and the complex, fuzzy classifier to calculate the appropriate class for the pattern if they don't.

Although [9] and [12] present well formed ideas about the combination of classifiers to improve the error rates of character recognition, the best results in [9] are dependent on a (relatively) high reject threshold, so the results of $< 0.2\%$ error rate are only attained when the reject threshold is set to 15%.

Even though the results in [9] are compared to realistic alternatives, and achieve better results through the method presented, one area of research could be to find a way of reducing this threshold without incurring such a large change in the error rate.

The work in [12] shows how the BKS-Method is an optimal combination of classifiers. However, this argument only escalates to an optimal total classifier when the underlying classifiers are also optimal. The work shows that for lower thresholds (in this case the Substitution Rate), the combination methods it is comparable to are just as effective, implying that this method should only be used in situations where the substitution rate is required to be high.

An area of research could be finding a way of taking the ideas from [14] of splitting the data based on how confident we are, and only using the BKS-Method on the ambiguous data, using one of the simpler methods for the easier classifications.

So, although reliable recognition of numerical characters is far from impossible, it is made much easier through the combination of complementary classifiers, due to how much easier it is to develop several simple classifiers as opposed to one quite complex classifier. It has been shown that combining classifiers in the general case can lead to optimality, provided the classifiers are optimal. It has also been shown that although the thresholds associated with classifiers are often linked, it is certainly an interesting area of research to identify whether this link is concrete, or could be avoided in certain situations. Another interesting area of research could be the combination of sub-optimal classifiers with a sub-optimal combination to see if it could be close enough to the optimal one for a real-world application.

Report comprises 1457 words as counted with *TeXCount*, excluding the bibliography.

References

[1] J. Paul Budnik, "Intuition as pattern recognition." [Online]. Available: http://www.mtnmath.com/whatrh/node106.html Last Visited: 25 Jan 2011.

[2] D. Baldwin, "The future and its impact." [Online]. Available: http://www.wayfinding.net/futimpt.htm Last Visited: 25 Jan 2011.

[3] Various, "Pattern recognition (psychology)." [Online]. Available: http://en.wikipedia.org/wiki/Pattern_recognition_(psychology) Last Visited: 25 Jan 2011.

[4] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.

[5] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, September 1995.

[6] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, January 1967.

[7] M. Hellman, "The nearest neighbor classification rule with a reject option," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 6, no. 3, pp. 179 –185, 1970.

[8] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, pp. 226–239, March 1998.

[9] J. Cao, M. Ahmadi, and M. Shridhar, "Recognition of handwritten numerals with multiple feature and multistage classifier," *Pattern Recognition*, vol. 28, no. 2, pp. 153–160, 1995.

[10] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, "Modified quadratic discriminant functions and the application to chinese character recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, pp. 149–153, January 1987.

[11] G. Ball and D. Hall, "A clustering technique for summarizing multivariate data," *Behav. Sci*, vol. 12, no. 2, pp. 153–5, March 1967.

[12] Y. S. Huang and C. Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, pp. 90–94, January 1995.

[13] Y. Huang and C. Suen, "An optimal method of combining multiple experts for handwritten numerical recognition," *Pre-Proc. International Workshop on Frontiers in Handwriting Recognition*, pp. 11–20, 1993.

[14] N. E. Mitrakis, J. B. Theocharis, and V. Petridis, "A multilayered neuro-fuzzy classifier with self-organizing properties," *Fuzzy Sets Syst.*, vol. 159, pp. 3132–3159, December 2008.