

# Biscuit Bake Assessment by an Artificial Neural Network

Jeffrey C.H. Yeh, and Leonard G.C. Hamey  
Department of Computing  
Macquarie University, NSW 2109, Australia

## Abstract

A prototype artificial neural network system for assessing the bake level of biscuits has been implemented. We present performance results and compare the neural network approach with a statistical method and the performance of the trained inspector. The neural network system performs comparably with the other methods.

## 1 Introduction

Inspection of baked products is very important for food manufacturers, as it ensures correct taste, texture and appearance. This task is normally performed by trained inspectors who examine the product and report unacceptable product. Human inspectors, however, provide subjective judgements that are prone to both short-term and long-term variations. Digital image processing systems, in contrast, provide objective appearance assessments. When digital image processing is combined with artificial neural networks (ANNs), the resulting system has the potential to learn objective assessment criteria from presentations of acceptable and unacceptable product samples. The experience of a trained inspector can therefore be captured in a machine inspection system, with the benefits of short-term and long-term consistency and reduced operating cost. The trained ANN may be deployed at the immediate post-production inspection point, providing for direct production control. Recently, the food industry has turned to artificial neural networks for product inspection with promising results [4, 5, 7].

We describe a prototype ANN system for the inspection of bake level in biscuits, as indicated by colour development with exposure to heat. In our experiments, one specific product was chosen (figure 1). The product is characterised by regions of high bake colour where a flakey thin blister forms on the top of the biscuit, and regions of low bake colour where blisters do not occur. The positions of the blisters are unpredictable. The inputs to the ANN are preprocessed intensity histograms of the sample images, and the network is trained to as-

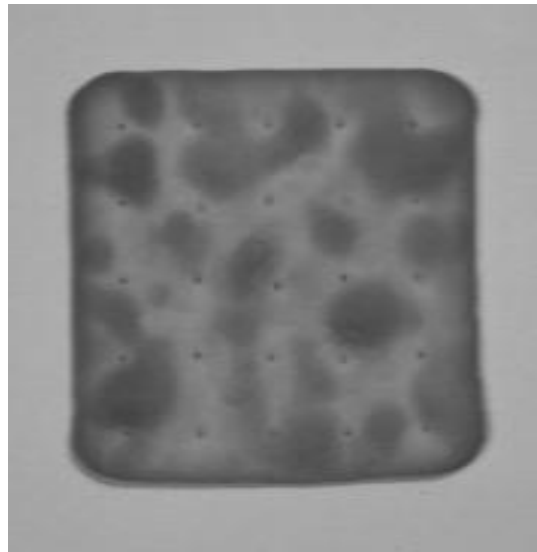


Figure 1: Product to be classified.

sess the product bake level based upon classification of the samples by the trained inspector.

## 2 Sample Preparation

Ninety biscuit samples were collected. Of these samples, thirty were nominally correctly baked, thirty underbaked and thirty overbaked. The samples were digitally imaged after centring them upon a white background (figure 1). Illumination was supplied by two 40W incandescent lamps, one placed to either side of the sample to minimise shadow effects. To eliminate the effect of illumination variation due to the use of AC power, the images were calibrated by a linear transformation of pixel values based on the measured intensity of the white background in each image. An intensity histogram was then produced for each sample image[1].

After imaging, the samples were classified by a trained inspector who separated them into three classes representing underbake, correct bake and overbake. These classification experiments were performed 10

times. As expected given the subjective nature of the classification task, the inspector’s performance was not totally repeatable. Each sample was therefore assigned as its *preferred class* the classification most frequently given to it by the inspector. The misclassification rate of the inspector was assessed relative to this preferred class. Samples that were misclassified by the inspector at least once are labelled as “misclassified” samples while those which were never misclassified are considered “firm”.

### 3 Network Topology

An ANN’s input size is important for its generalization performance. An insufficient number of input nodes can hide the necessary information too deeply in the input or obscure it with other properties [3, pp76–77]. An excessive number of input nodes can provide too many degrees of freedom, allowing the neural network to overfit the training samples and produce poor results on unseen data [6, pp102–107]. To determine the most suitable number of input nodes, we experimented with networks employing 4, 8, 16, 24, 32, 40, 48 and 56 input nodes. We reduced the size of the intensity histograms to the number of ANN input nodes by low-pass filtering the histograms with a Gaussian filter and subsampling them at intervals of two standard deviations. For each network topology, we report in table 1 the number of input nodes which provided the best performance on the cross-validation data set.

The choice of hidden and output units also significantly affects the learning performance of an ANN. The selection of a suitable topology is still regarded by many as an art. We investigated four alternative neural network topologies. The  $N-1$  topology had  $N$  input units and one output unit. The  $N-2$  topology had  $N$  input units and two output units while the  $N-3$  topology had three output units. We also investigated a topology with  $N$  input units, two hidden units and one output unit, represented as  $N-2-1$ . This topology did not employ short-cut connections between the input and output units.

The  $N-3$  topology used the binary patterns 100, 010 and 001 to represent underbake, correct bake and overbake respectively. The  $N-2$  topology employed the patterns 10, 00 and 01 to represent underbake, correct bake and overbake respectively.

The topologies with a single output unit were trained to represent the bake level assessment using an output value of 0.0 for underbake, 0.5 for correct bake and 1.0 for overbake. Two different methods were used to compute the target output values. In the *class targets* method, the target values were specified as 0.0, 0.5 or 1.0 based upon the preferred class for each sample. In

the alternative *graded targets* method, the target values were specified as the average inspector classification, where each classification was coded using 0.0 for underbake, 0.5 for correct bake and 1.0 for overbake.

### 4 Network Training

For training and testing of the artificial neural networks, the Aspirin simulator V6.0 [2] was employed. We enhanced the simulator to perform a form of cross-validation [6, pp31–33]. After every 1200 training iterations (representing 400 complete presentations of the training data) the RMS error of the network was computed on the evaluation data set. The weights of the network with the lowest RMS error were saved and used as the final trained network. In order to ensure that the global minimum of the RMS error was achieved, training was continued until 300 successive evaluations produced a larger RMS error than the best value.

All of our experiments employed straight-forward back-propagation learning and were conducted with a learning rate of 0.05 and a momentum of 0.95. A learning threshold of 0.00005 was employed to avoid training on patterns which were successfully learnt.

The 90 data samples were randomly divided into three groups of 30 samples each. To ensure balanced results, six training experiments were performed for each neural network configuration. In each training experiment, one of the groups was used to train the network, a second group was used for cross-validation during training and the remaining group was used to test the trained neural network. The six experiments represent the six permutations of the three data groups. The results presented in the next section are the combined performance results from the six experiments.

### 5 Network Performance

Table 1 presents the observed neural network performance. For each network topology, performance results are presented for the input size which obtained the best RMS error on the cross-validation data. For comparison with the 32–1 and 40–1 topologies, performance results for the 8–1 topology are also presented.

The “Targets” column of the table indicates the type of target data that was used for training. For topologies with a single output unit, class or graded target data were used (see section 3). For network topologies with multiple output units, binary patterns were used to represent the bake class.

Error statistics are reported for both the cross-validation data sets and the test data sets. The error measures on the test data set provide an unbiased estimate of the error on unseen data, while the error mea-

Table 1: Neural Network Performance

Topology	Targets	Error measure	Cross validation RMS error	Cross validation error count	Cross validation error rate (%)	Test RMS error	Test error count	Test error rate (%)
32-1	class	class	0.151	20	11	0.156	19	11
40-1	graded	class	0.152	20	11	0.155	25	14
8-2-1	class	class	0.123	13	7	0.137	18	10
8-2-1	graded	class	0.123	15	8	0.126	15	8
8-2	binary	binary	0.185	16	9	0.194	16	9
24-3	binary	binary	0.236	33	18	0.252	36	20
24-3	binary	wta		17	9		20	11
32-1	class	graded	0.119	10	6	0.123	10	6
40-1	graded	graded	0.118	8	4	0.120	10	6
8-2-1	class	graded	0.096	6	3	0.105	10	6
8-2-1	graded	graded	0.090	8	4	0.094	8	4
8-1	class	class	0.152	21	12	0.158	20	11
8-1	graded	graded	0.122	13	7	0.129	15	8

sures on the cross-validation data set are the appropriate basis for selection of an optimal network topology. The test error counts are the number of errors observed in 180 tests performed (90 samples, each tested twice).

The RMS and error count statistics have been computed using different error measures. For networks with one output unit, the *class* error measure counts an error if the network’s output differs from the preferred class value (0.0, 0.5 or 1.0) by more than 0.25. The *graded* error measure, also used for single-output networks, counts an error if the network’s output differs from the graded target value by more than 0.25. The graded error measure consistently produces a lower error rate.

For networks with multiple output nodes, the *binary* error measure counts an error if the networks output differs from the desired target value by more than 0.5. An alternative error measure for the three-output case is a winner-take-all (wta) method where the network’s most active output node is taken as its classification.

The RMS error statistic can be used to compare network topologies which employ the same target data and have the same number of output units. For comparison between networks which employ different output representations, however, it is necessary to consider the count of erroneous classifications. This count is best represented by the class error measure for single output networks, the binary error measure for two output networks and the winner-take-all error measure for three output networks. Since a topology is being selected, the cross-validation error count should be used rather than the test data error count. On this basis, the 8-2-1 topology trained with class targets produces the

best performance with an error rate of 7%. The 8-2-1 topology trained with graded targets achieves nearly the same performance and the 8-2 topology is a close third. The 24-3 topology performs marginally less well when evaluated with the winner-take-all error measure, and the topologies with a single output unit and no hidden units perform worst of all.

It is interesting to note that the  $N-1$  topologies perform better with a larger number of input units, allowing the network to reflect more fine detail of the intensity histogram structure. The resulting networks generalize better than the corresponding 8-1 topologies as shown by the test error statistics.

Training single output topologies with graded targets appears to be of little or no benefit for classification performance of the trained network. It also appears to have little effect upon the ability of the network to produce a graded bake assessment, as shown in the second part of table 1. This suggests that the single-output networks are producing a useful graded response, even when trained with class targets.

## 6 Performance Comparison

Table 2 reports the test error of the trained networks with the best cross-validation error rate compared to the performance of a statistical method [1] and the performance of the human inspector. For each method, the error rates are reported separately for the 62 “firm” samples and the 28 samples which were misclassified at least once by the inspector. The misclassified samples are expected to have a higher error rate because

Table 2: Performance Comparison of Different Approaches

Approach	Targets	Error Criterion	Test error count (/180)	Test error rate (%)	Firm sample error (%)	Misclassified sample error (%)
8-2-1	class	class	18	10	4	23
8-2-1	graded	class	15	8	3	20
8-2	binary	binary	16	9	1	27
Statistical			24	13	3	36
Human				7		

they are likely to lie near the boundary between two classes. Indeed, all the machine inspection methods produce much higher error rates for the misclassified samples and their error rates for the “firm” samples are suitably small.

In comparison with the human inspector, the neural networks are performing almost equally well. Although the neural network error rate of 9% is a little worse than the inspector’s error rate of 7%, the difference is not statistically significant.

The neural networks have outperformed the statistical method, indicating that a neural network approach may well be superior to a straight-forward statistical method. The statistical performance figure reported here (13% error rate) is significantly worse than that reported in [1] (6% error rate). The latter statistic was based on a leave-one-out cross-validation, where 89 samples were analysed to classify the remaining sample. Such a method is not comparable with the performance of a neural network trained on only 30 samples. The performance result reported above was obtained by analysing 30 samples and testing the performance on the remaining 60 samples, with the experiment repeated for each set of 30 samples.

## 7 Conclusion

We have investigated the application of artificial neural networks to the assessment of bake level in biscuits. We have shown that suitably trained networks with appropriate topologies perform comparably with the trained human inspector. The neural networks were trained with only 30 data samples, a small number for such a task. It is expected that a larger data set would improve the neural network performance.

Our experiments show that artificial neural networks are suitable as a method of automated bake assessment. Future work will investigate their application to other products.

## 8 Acknowledgement

This research was supported by Arnott’s Biscuits Ltd. The authors especially thank Tas Westcott and Anne Watson for assistance in this research.

## References

- [1] Leonard G. C. Hamey, Annesley J. Watson, and C. Tasman Westcott. Machine inspection of biscuit bake. In K. K. Fung and A. Ginige, editors, *Proceedings of Digital Image Computing Techniques and Applications*, pages 124–129. Australian Pattern Recognition Society, 1993.
- [2] Russell R. Leighton. The aspirin/migraines neural network software: User’s manual. Technical Report MP-91W00050, The MITRE Corporation, 1992.
- [3] Müller and Reinhardt. *Neural Networks: an Introduction*. Springer-Verlag, 1990.
- [4] Lewis Reid. Neural networks: Is there a role in food? *Food Manufacture*, pages 41–42, February 1992.
- [5] Bob Sperber. Prime time for machine vision. *Food Processing*, 53(10):19, 21, 24–25, October 1992.
- [6] Scholom M. Weiss and Casimir A. Kulikowski. *Computer Systems that Learn*. Morgan Kaufmann Publishers, 1991.
- [7] A. Dale Whittaker, Bo Soon Park, James Darrell McCauley, and Yanbo Huang. Ultrasonic signal classification for beef quality grading through neural networks. In *Proceedings of the Automated Agriculture for the 21st Century Symposium*, 1991.