Combining Methods to Stabilize and Increase Performance of Neural Network-Based Classifiers

Fabricio A. Breve  
Moacir P. Ponti Jr.  
Nelson D. A. Mascarenhas

fabricio@dc.ufscar.br  
moacir@dc.ufscar.br  
nelson@dc.ufscar.br

DC – Departamento de Computação  
UFSCar - Universidade Federal de São Carlos, São Paulo, SP, Brasil
Goals

- Recognize materials in multispectral images (obtained with a tomograph scanner) using a Neural Network based classifier (Multilayer Perceptron)
- Investigate classifier combiner techniques in order to improve performance
Summary

- Image Acquisition
- Classification Methods
- Evaluation
- Experiments
- Results
- Conclusions
- Future Works
Image Acquisition

- First generation Computerized Tomograph developed by Embrapa in order to explore applications in soil science
  - X-Ray and γ-ray fixed sources
  - Object being studied is rotated and translated
Image Acquisition

- Phantom built with materials found in soil
- Plexiglass support
- 4 Cylinders containing: Aluminum, Water, Phosphorus and Calcium
Image Acquisition

- **X-ray sources:**
  - 40 keV
  - 85 keV

- **γ-ray sources**
  - 60 keV (Americium)
  - 662 keV (Cesium)

- 65x65 pixels
- 3 seconds of exposure
- 256 levels of gray
- High level of noise
Classification Methods

- **Multilayer Perceptron**
  - Neural-Network Based Classifier
  - One Input Layer, one or more hidden layers, one Output Layer
  - Training: Error Backpropagation Algorithm
  - Hidden layers and output layers with Sigmoid functions

\[ g(a) \equiv \frac{1}{1 + \exp(-a)} \]
Classification Methods

- Bagging
  - Bootstrap AGGretatING
  - Bootstrap sets built randomly from the original training set using substitution
  - Each bootstrap set trains a classifier
  - Outputs are combined using majority voting
  - Requires the base classifier to be unstable
    - minor differences in the training set can lead to major changes in the classifier
Classification Methods

- **Decision Templates**
  - Continuous-valued outputs from each classifier with a different initialization for a given sample are used to build a decision profile.
  - The Decision Templates are the mean over all the decision profile from each training sample for each class.
  - The label of a test sample is chosen by comparing its decision profile with each decision template and choosing the most similar one.

\[
DP(x) = \begin{bmatrix}
  d_{1,1}(x) & \cdots & d_{1,j}(x) & \cdots & d_{1,c}(x) \\
  \vdots & & \vdots & & \vdots \\
  d_{i,1}(x) & \cdots & d_{i,j}(x) & \cdots & d_{i,c}(x) \\
  \vdots & & \vdots & & \vdots \\
  d_{L,1}(x) & \cdots & d_{L,j}(x) & \cdots & d_{L,c}(x)
\end{bmatrix}
\]
Classification Methods

- Dempster-Shafer
  - Based on the Evidence Theory, a way to represent cognitive knowledge
  - It’s like the Decision Templates method, but for each test sample we calculate the proximity between the decision template and the output of each classifier.
  - These are used to calculate the belief degree for every class.
  - At last the final degrees of support of each test sample for each class are calculated from the belief degrees.
Evaluation

- **Hold-Out**
  - Splits the set of available data in two halves:
    - Training set
    - Testing set
  - It’s a fast testing scheme, as required by Multilayer Perceptron

- **Kappa Coefficient**
  - Measures the agreement rating between the classification of the test samples and their true class
Experiments

- 480 samples (80 samples from each of the 6 class):
  - Aluminum
  - Water
  - Phosphorus
  - Calcium
  - Plexiglass
  - Background

- 240 samples (40 from each class) for training
- 240 samples for testing
Experiments

- Networks with 2 to 15 units in one single hidden layer
- Every experiment for all classification methods were executed 100 times
- Bagging using means instead of majority vote to take advantage of continuous-valued output
- All experiments with classifier combiners used 10 base classifiers with different initializations
Results – Estimated Error

![Graph showing estimated error with units in the hidden layer on the x-axis and estimated error on the y-axis. The graph compares Single, Bagging, DT, and DS models. The error decreases as the number of units in the hidden layer increases.]
## Results – Estimated Error

<table>
<thead>
<tr>
<th>Units in the hidden layer</th>
<th>Single Classifier</th>
<th>Bagging</th>
<th>DT</th>
<th>DS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.5720</td>
<td>0.3675</td>
<td>0.0349</td>
<td>0.0613</td>
</tr>
<tr>
<td>3</td>
<td>0.2689</td>
<td>0.0493</td>
<td>0.0163</td>
<td>0.0275</td>
</tr>
<tr>
<td>4</td>
<td>0.1318</td>
<td>0.0200</td>
<td>0.0141</td>
<td>0.0177</td>
</tr>
<tr>
<td>5</td>
<td>0.0976</td>
<td>0.0170</td>
<td>0.0123</td>
<td>0.0151</td>
</tr>
<tr>
<td>6</td>
<td>0.0741</td>
<td>0.0168</td>
<td></td>
<td>0.0139</td>
</tr>
<tr>
<td>7</td>
<td>0.0681</td>
<td>0.0175</td>
<td>0.0129</td>
<td>0.0138</td>
</tr>
<tr>
<td>8</td>
<td>0.0636</td>
<td>0.0179</td>
<td>0.0130</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.0511</td>
<td>0.0190</td>
<td>0.0134</td>
<td>0.0138</td>
</tr>
<tr>
<td>10</td>
<td>0.0570</td>
<td>0.0190</td>
<td>0.0135</td>
<td>0.0139</td>
</tr>
<tr>
<td>11</td>
<td>0.0497</td>
<td>0.0191</td>
<td>0.0136</td>
<td>0.0143</td>
</tr>
<tr>
<td>12</td>
<td>0.0603</td>
<td>0.0203</td>
<td>0.0136</td>
<td>0.0143</td>
</tr>
<tr>
<td>13</td>
<td>0.0525</td>
<td>0.0196</td>
<td>0.0137</td>
<td>0.0145</td>
</tr>
<tr>
<td>14</td>
<td>0.0477</td>
<td>0.0204</td>
<td>0.0140</td>
<td>0.0151</td>
</tr>
<tr>
<td>15</td>
<td><strong>0.0470</strong></td>
<td>0.0210</td>
<td>0.0143</td>
<td>0.0150</td>
</tr>
</tbody>
</table>
Results – Kappa Coefficient

![Graph showing Kappa Coefficient for different units in the hidden layer, with lines for Single, Bagging, DT, and DS.]
## Results – Kappa Coefficient

<table>
<thead>
<tr>
<th>Units in the hidden layer</th>
<th>Single Classifier</th>
<th>Bagging</th>
<th>DT</th>
<th>DS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.3137</td>
<td>0.5591</td>
<td>0.9581</td>
<td>0.9265</td>
</tr>
<tr>
<td>3</td>
<td>0.6773</td>
<td>0.9409</td>
<td>0.9805</td>
<td>0.9671</td>
</tr>
<tr>
<td>4</td>
<td>0.8419</td>
<td>0.9760</td>
<td>0.9831</td>
<td>0.9788</td>
</tr>
<tr>
<td>5</td>
<td>0.8829</td>
<td>0.9796</td>
<td>0.9853</td>
<td>0.9819</td>
</tr>
<tr>
<td>6</td>
<td>0.9111</td>
<td><strong>0.9799</strong></td>
<td>0.9848</td>
<td>0.9833</td>
</tr>
<tr>
<td>7</td>
<td>0.9183</td>
<td>0.9790</td>
<td>0.9845</td>
<td>0.9835</td>
</tr>
<tr>
<td>8</td>
<td>0.9237</td>
<td>0.9786</td>
<td>0.9844</td>
<td><strong>0.9836</strong></td>
</tr>
<tr>
<td>9</td>
<td>0.9387</td>
<td>0.9773</td>
<td>0.9840</td>
<td>0.9835</td>
</tr>
<tr>
<td>10</td>
<td>0.9316</td>
<td>0.9773</td>
<td>0.9838</td>
<td>0.9833</td>
</tr>
<tr>
<td>11</td>
<td>0.9404</td>
<td>0.9771</td>
<td>0.9837</td>
<td>0.9829</td>
</tr>
<tr>
<td>12</td>
<td>0.9277</td>
<td>0.9757</td>
<td>0.9837</td>
<td>0.9829</td>
</tr>
<tr>
<td>13</td>
<td>0.9371</td>
<td>0.9765</td>
<td>0.9836</td>
<td>0.9827</td>
</tr>
<tr>
<td>14</td>
<td>0.9428</td>
<td>0.9756</td>
<td>0.9832</td>
<td>0.9819</td>
</tr>
<tr>
<td>15</td>
<td><strong>0.9436</strong></td>
<td>0.9748</td>
<td>0.9828</td>
<td>0.9821</td>
</tr>
</tbody>
</table>
Results – Thematic Images

Thematic images for the best classifier of each group (best case)

Single Classifier
Estimated Error: 0.0083
Kappa Coefficient: 0.9900

Bagging
Estimated Error: 0.0083
Kappa Coefficient: 0.9900

Decision Templates
Estimated Error: 0.0042
Kappa Coefficient: 0.9950

Dempster-Shafer
Estimated Error: 0.0125
Kappa Coefficient: 0.9850

Thematic images for the best classifier of each group (worst case)

Single Classifier
Estimated Error: 0.3417
Kappa Coefficient: 0.5900

Bagging
Estimated Error: 0.0250
Kappa Coefficient: 0.9700

Decision Templates
Estimated Error: 0.0208
Kappa Coefficient: 0.9750

Dempster-Shafer
Estimated Error: 0.0208
Kappa Coefficient: 0.9750

water       aluminum       phosphorus       calcium       plexiglass       background
Results – Standard Deviation

![Graph showing Standard Deviation vs Units in the hidden layer for different models (Single, Bagging, DT, DS). The graph illustrates the decrease in standard deviation as the number of units in the hidden layer increases for each model.](image-url)
Conclusions

- Classifier combination led to more stable classifiers, where even the worst case still delivers a good classification.
- The best results with all the combiners were achieved with few units in the hidden layer.
- Dempster-Shafer and Decision Templates combiners showed relatively good results no matter how many units there were in the hidden layer.
Conclusions

- Multilayer Perceptron based classifiers to identify materials on CT images is viable, even in images with high noise levels.
- The use of classifiers combiners led to more stable systems and minimized the effects of the unstable nature of the individual MLP classifiers.
Future Works

- experiments with more than 10 base classifiers to observe if the performance improvements could worth the extra time to train all the base classifiers

- a combination using Bagging techniques, but using Dempster-Shafer or Decision Templates as the combiner, instead of the mean combiner we used in our experiments
Acknowledgements

- Dr. Paulo E. Cruvinel for providing the multispectral images used in the experiments
- CAPES and FAPESP (grant n. 04/05316-7) for student scholarship
- This work was also partially supported by FAPESP Thematic Project 2002/07153-2
Combining Methods to Stabilize and Increase Performance of Neural Network-Based Classifiers

Fabricio A. Breve  
Moacir P. Ponti Jr.  
Nelson D. A. Mascarenhas

fabricio@dc.ufscar.br  
moacir@dc.ufscar.br  
nelson@dc.ufscar.br

DC – Departamento de Computação  
UFSCar - Universidade Federal de São Carlos, São Paulo, SP, Brasil