SUPERVISED FEATURE EMBEDDING FOR CLASSIFICATION BY LEARNING RANK-BASED NEIGHBORHOODS

Ghazaal Sheikhi, Hakan Altıncay
Department of Computer Engineering
Eastern Mediterranean University
Famagusta, North Cyprus, Mersin 10, Turkey
Abstract

• Proposed feature embedding method, representative learning of rank-based neighborhoods (RLRN) is inspired by the well-known word embedding technique, word2vec.

• The notion of context words in word2vec is extended into neighboring instances.

• Each sample is represented by a unique one-hot vector whereas its neighbors are encoded by several two-hot vectors.

• A feed-forward neural network with a continuous projection layer, learns the mapping from one-hot vectors to multiple two-hot vectors.

• The hidden layer determines the reduced subspace for the train samples.

• Experimental results confirm that the proposed method is effective in finding a discriminative representation of the features.
An example of neighborhood encoding in RLRN

<table>
<thead>
<tr>
<th>$f_1$</th>
<th>$f_2$</th>
<th>$f_3$</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>3</td>
<td>121</td>
<td>0</td>
</tr>
<tr>
<td>0.4</td>
<td>4</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>0.3</td>
<td>0</td>
<td>40</td>
<td>1</td>
</tr>
<tr>
<td>0.1</td>
<td>5</td>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>0.7</td>
<td>8</td>
<td>32</td>
<td>1</td>
</tr>
</tbody>
</table>

Neighborhood window size: $w = 2$

- Rank of $x_3$ in $f_1$ dimension is 5:
  Samples with ranks 3, 4, 5, 6, 7 are $x_3$’s neighbors

- Rank of $x_3$ in $f_2$ dimension is 2:
  Samples with ranks 1, 2, 3, 4 are $x_3$’s neighbors

- Rank of $x_3$ in $f_3$ dimension is 1:
  Samples with ranks 1, 2, 3 are $x_3$’s neighbors

$\alpha = (0, 0, 1, 0, 0, 0)$

$\mathbf{t}^1 = (1, 0, 1, 0, 0, 0)$

$\mathbf{t}^2 = (0, 1, 1, 0, 0, 0)$

$\mathbf{t}^3 = (0, 1, 1, 0, 0, 0)$

$\mathbf{t}^4 = (0, 1, 1, 0, 0, 0)$

$w = 2$ means if the rank difference is less than or equal to 2, the samples are assumed neighbors given that they belong to the same class.
The neural network architecture of RLRN

- One-hot encoded samples
  - $\alpha_1$
  - $\alpha_{N-1}$
  - $\alpha_N$

- Two-hot encoded neighbors
  - Neighbor 1
  - Neighbor $k$

Input layer → Projection layer

$W_{d\times N} = \{w_{ij}\}$

Output layer

$W_{N\times d} = \{w'_{ij}\}$
The neural network architecture of RLRN

The weight matrix between the input and the hidden layer

$$h = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_d \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{d1} & w_{d2} & \cdots & w_{dN} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}$$

$\alpha$ is a one-hot vector with its $\alpha_n^*$ equal to 1 and its other elements are 0.

The matrix multiplication is thus unnecessary:

$$h = \begin{bmatrix} w_{1n^*} \\ w_{2n^*} \\ \vdots \\ w_{dn^*} \end{bmatrix} = w_{n^*}$$
The neural network architecture of RLRN

The weight matrix between the hidden layer and the output layer

\[ u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} w'_{11} & w'_{12} & \cdots & w'_{1N} \\ w'_{21} & w'_{22} & \cdots & w'_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w'_{d1} & w'_{d2} & \cdots & w'_{dN} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{bmatrix} \]

\[ u = W'.w_n^* \]

The elements of the output vector

\[ y_i = \frac{1}{1 + e^{-u_i}} \]
Updating weights

The error on the \( i \)th output node of the \( k \)th neighbor

\[
e_i^\kappa = -t_i^\kappa \cdot \log(y_i)
\]

The sum operator only takes values for the two-hot elements of \( t_i^k \) i.e. \( t_i^k\)

\[
e_i = -\sum_{\kappa \in \kappa^*} t_i^\kappa \cdot (u_i - \log(1 + e^{u_i}))
\]

The update formulas for \( W' \) and \( W \)

\[
w'_{ij}(t + 1) = w'_{ij}(t) + \eta \cdot \sum_{\kappa \in \kappa^*} t_i^\kappa \cdot (1 - y_i) \cdot h_j
\]

\[
w_{ij}(t + 1) = w_{ij}(t) + \eta \cdot \sum_{\kappa \in \kappa^*} t_i^\kappa \cdot (1 - y_i) \cdot w'_{ij} \cdot \alpha_i
\]
Experiments

10-fold cross validation

Target dimension \( d = \{2, 3, ..., \min(10, D)\} \)

SVM and kNN

Accuracy and AUC

**Data sets and their characteristics**

<table>
<thead>
<tr>
<th>Name</th>
<th>Sample (N)</th>
<th>Features (D)</th>
<th>Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris</td>
<td>150</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Parkinsons</td>
<td>195</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>Seeds</td>
<td>210</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Breast Cancer</td>
<td>116</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Breast Tissue</td>
<td>106</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>Wine</td>
<td>178</td>
<td>13</td>
<td>3</td>
</tr>
<tr>
<td>Sonar</td>
<td>208</td>
<td>60</td>
<td>2</td>
</tr>
<tr>
<td>SPECTF</td>
<td>267</td>
<td>44</td>
<td>2</td>
</tr>
<tr>
<td>Leaf</td>
<td>340</td>
<td>14</td>
<td>30</td>
</tr>
</tbody>
</table>
Methods for comparison

• Unsupervised
  1. PCA
  2. fastICA

• Unsupervised
  3. AMMC (Adaptive Maximum Margin Criterion)
  4. DSPP (Discriminative Sparsity Preserving Projection)
  5. MMC (Maximum Margin Criterion)
  6. MSD (Maximum Scatter Difference)
  7. DNE (Discriminant Neighborhood Embedding)
  8. LSDA (Locality Sensitive Discriminant Analysis)
  9. MMP (Maximum Margin Projection)
 10. MFA (Marginal Fisher Analysis)
The highest improvement with respect to no embedding

The highest improvement in AUC

RLRN
PCA
fastICA
MMP
AMMC
DSPP
MMC
MSD
DNE
LSDA
MFA

The highest improvement in Accuracy

RLRN
PCA
fastICA
MMP
AMMC
DSPP
MMC
MSD
DNE
LSDA
MFA

SVM
3NN
Mean
Mean accuracy of SVM and 3NN for different dimensions of embedded subspace
Mean AUC of SVM and 3NN for different dimensions of embedded subspace
The highest accuracy (mean of SVM and 3NN) for different settings of neighborhood parameter.
Methods reaching the highest accuracy and AUC for each data set

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Winner Accuracy</th>
<th>Winner AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris</td>
<td>fastICA</td>
<td>MSD</td>
</tr>
<tr>
<td>Parkinsons</td>
<td>RLRN</td>
<td>RLRN</td>
</tr>
<tr>
<td>Seeds</td>
<td>fastICA</td>
<td>MMC</td>
</tr>
<tr>
<td>Breast Cancer</td>
<td>RLRN</td>
<td>RLRN</td>
</tr>
<tr>
<td>Breast Tissue</td>
<td>DSPP</td>
<td>MFA</td>
</tr>
<tr>
<td>Glass</td>
<td>MSD</td>
<td>RLRN</td>
</tr>
<tr>
<td>Wine</td>
<td>RLRN</td>
<td>RLRN</td>
</tr>
<tr>
<td>Sonar</td>
<td>RLRN</td>
<td>RLRN</td>
</tr>
<tr>
<td>SPECTF</td>
<td>PCA</td>
<td>RLRN</td>
</tr>
<tr>
<td>Leaf</td>
<td>MSD</td>
<td>DSPP</td>
</tr>
</tbody>
</table>
The highest AUC (mean of SVM and 3NN) for different settings of neighborhood parameter.
Conclusions

• Inspired by the well-known word2vec, RLRN employs a one/two-hot vector encoding and a FFNN.

• The FFNN learns the homogeneous neighborhoods of data points.

• A discriminative lower-dimensional subspace is generated in the hidden layer.

• RLRN is significantly superior to state-of-the-art in terms of accuracy and AUC.

✓ As a possible future direction, adopting the proposed method to large data sets can be considered.
THANKS FOR YOUR ATTENTION