Complementary Learning Fuzzy Neural Network: An Approach to Imbalanced Dataset

T. Z. Tan, G. S. Ng, and C. Quek

Abstract—Imbalanced dataset is a phenomenon seen in many real-life applications, especially in medical field. The conventional computational intelligence algorithms cannot effectively handle the imbalanced data because they are designed for balanced data distribution. Complementary learning fuzzy neural network is proposed as one of the approach for learning imbalanced dataset. It is shown empirically and theoretically that the effects of imbalanced dataset are minimal in this class of neuro-fuzzy system.

I. INTRODUCTION

Imbalanced data is a frequent encountered phenomenon in real life. In this kind of data, one class is significantly outnumbered by other classes. It is also known as having a skew distribution. Many real world data exhibit this characteristic, especially, medical data. Oftentimes, the targeted class (known as positive) is significantly lesser than that of the non-targeted class. As a result, classification is difficult because most of the conventional computational intelligence methods are not designed with the ability to handle such imbalance. They tend to classify all samples to the majority class [1]. Yet, it is important to derive an algorithm to learn imbalance dataset as it is prevalent in real world data.

Formally, given a dataset \( D \) made up of \( T \) sample pairs

\[ D = \bigcup_{r=1}^{T} \{ (x, y) \} \]

it can be split into positive and negative,

\[ D = \{ D^+ \cup D^- \} \]

Looking at a two-class problem, positive is the data associated with \( y = 1 \), symbolizing the targeted class, whereas negative is the data associated with \( y = 0 \).

Likewise, the input \( x \) can be categorized into whether they are in positive or negative regions, i.e., \( x \cap X^+ \) or \( x \cap X^- \).

An imbalanced dataset \( D_{imbalanced} \) is the one with

\[ |D^+| << |D^-| \]

where << denotes significantly lesser, and \( |\cdot| \) denotes the cardinality.

Popular methods in medical area such as Multilayer Perceptron (MLP), Radial Basis Function (RBF), and Linear Discriminant Analysis (LDA) cannot effectively classify imbalanced dataset because they learn the dataset based on minimization of the classification accuracy, which misrepresented the real scenario. Furthermore, they do not take into account the error costs of the classes [2][3][4]. When learning imbalanced dataset, their performances are not guaranteed and may perform very badly. Decision tree C4.5, Support Vector Machine (SVM), and some neural models could not escape [5][6][7] from this as well. Thus, steps that can enable these methods to learn imbalanced dataset are required. Alternatively, new methods have to be devised.

Many endeavors were made toward this end. In general, the approaches for imbalanced dataset can be grouped into two: data processing (external) and algorithmic (internal) approaches. The majority of the techniques seek to improve the minority class prediction [8]. The former approach is the method that alters the class distribution, which includes down-sampling the majority class, and up-sampling the minority class. In down sampling approaches, the majority class instances are removed [9]. The bagging technique, which train multi-classifiers on different subsets of the minority class belongs to this category. Apart from this, some propose the use of clustering prior to the system training. Subsequently, the system is trained with the clusters representing the majority class [10]. Unfortunately, despite the advantage of being simple and of reducing large number of training instances, down-sampling methods suffer information loss. This is because majority of the training instances are discarded to make the distribution balance. On the other hand, up-sampling involves the introduction of replicated, synthetic or interpolated samples of the minority class. However, up-sampling by interpolation relies on the assumption that neighborhood of positive instance will still be positive, which may not be true [9]. Above and beyond that, up-sampling will introduce noise, especially when the chosen instance is noisy, the undesirable effect will be amplified [7]. For data preprocessing methods, there is another family of approach that uses weighting on the minority samples. However, these approaches alter the class distribution. This ignores the real data distribution of the positive and negative classes, which could be useful for classification.

The later approach is the algorithmic technique, which embraces methods from rules weighting [11], decision threshold tuning [12], to cost matrices modification [12], asymmetrical cost function [12], hyperplane readjustment [13], etc. These methods are useful to alleviate the effects of
imbalanced dataset. However, when compared to sampling method, they lack the rigorous and systematic treatment on the dataset. Furthermore, since they are specific, they are less versatile when compared against sampling methods [5][12].

Owing to that, an alternative to imbalanced dataset, Complementary Learning Fuzzy Neural Network (CLFNN), is proposed in this work. Neural fuzzy model is employed as it is found to be less variant to class distribution and less sensitive to the data imbalance [2]. The adoption of fuzzy logic allows the system to tolerate uncertainty in the data and hence, reduces the effect of data imbalance. In contrast to standard methods, CLFNN requires no data preprocessing prior to its training. Unlike, sampling methods, CLFNN does not assume or alter the class distributions. CLFNN does not introduce bias to favor the minority class as well. Instead, CLFNN maintain a segregated positive and negative rulebase to describe the positive and negative classes. The construction of the rulebase is segregated, and this alleviates the influence of the majority class to the minority class. Furthermore, CLFNN inference process exploits the lateral inhibition between positive and negative knowledge. Thus, CLFNN could potentially offer good performance in pattern recognition, even if it is an imbalanced dataset. This work seeks to assess the feasibility of CLFNN as an alternative for the target class; (2) neutral learning system, where there is no concept of positive and negative; (3) complementary learning system, where the system generates knowledge based on positive and negative classes, along with exploitation of the relationship between positive and negative examples.

Formally, CLFNN is a 9-tuple, \( \langle X, Y, D, A, R, B, l, s, p \rangle \). The definition and description of each CLFNN component is given in the following.

\( X \) An observation/event/object; is a set of attributes \( x = (x_1, \ldots, x_l) \); \( X \in U_1 \times U_2 \times \ldots \times U_l = U \), where \( U \) denotes the input space.

\( Y \) Outcome(s) associated with \( X \); \( Y = Y^+ \cup Y^- \), where \( Y^+ \) and \( Y^- \) denote positive and negative outcomes, respectively; \( Y \in V \), where \( V \) represents the output space, and \( V = V_1 \times V_2 \times \ldots \times V_M \), where \( M \) is the total number of possible outcomes. Note that, most of the time, these outcomes are often mutually exclusive or disjoint (for classification problem), that is, \( \forall_m \in \Lambda, (m, n) \in M, Y_m \cap Y_n = \emptyset \). This is to say, if \( Y^+ = \{Y_m\} \), then \( Y^- = Y - Y^+ = \bigcup_{n \in M, n \neq m} Y_n \).

\( D \) Source knowledge/data; is a set of observations \( X \) and its corresponding outcome \( Y \). That is, \( D = \bigcup_{t=1}^{T} \langle X, Y \rangle \), where \( T = |D| \) total number of observation and outcome pairs. Since \( Y = Y^+ \cup Y^- \), \( D \) can be divided into positive and negative data, \( D = D^+ \cup D^- \).

\( A \) The behaviors; is a set of knowledge describing the behavioral relationships amongst observations \( X \) in \( D \). These relationships can be represented by linguistic, logistic or mathematical term.

\( B \) The behaviors; is a set of knowledge describing the behavioral relationships amongst observations \( Y \) in \( D \).

\( R \) Rules; is the structural relationship between \( A \) and \( B \). In other words, \( R \) represents the knowledge learnt from the data \( D \), linking the inputs to the correct outcomes. Therefore, the membership function of \( R \), \( \mu_R \), maps \( A \) to \( B \), that is, \( \mu_R : A \rightarrow B \). As a result, \( R = R^+ \cup R^- \), where \( R^+ \) contains positive knowledge constructed from \( D^+ \) and \( R^- \) contains negative knowledge constructed from \( D^- \).

\( l \) Learning function; constructs the CLFNN based on \( D \). That is, \( l : D \rightarrow \langle A, B, R \rangle \). It comprises two complementary learning functions: structure learning, \( s \), and parameter learning, \( p \). The two can be carried out concurrently, or sequentially (\( s \), then \( p \)), or independently.

\( s \) Structure learning; constructs the network. It can be any clustering algorithm that derives the structures from data autonomously. It constructs the fuzzy sets and memory elements that describe the underlying data. It
adds new node if the existing knowledge is not sufficient to describe the data. The process ends when it deems that the structure of the network is appropriate to reflect the data D. Since s is performed on X and Y spaces, S can be written as the confluence of

\[ s_X : X \rightarrow A, s_Y : Y \rightarrow B, s_R : A \rightarrow B. \]

Normally, the same structure learning is chosen for \( s_X \) and \( s_Y \). Function \( s_R \) relates the input space \( A \) to output space \( B \). Weights can be applied to the linkages if there is a need to. Since \( s \) is responsible for implementing the positive and negative learning architecture, \( s \) constructs positive and negative knowledge separately. That is,

\[
\begin{align*}
  s_X : & \quad X^+ \rightarrow A^+, & \quad X^- \rightarrow A^- \\
  s_Y : & \quad Y^+ \rightarrow B^+, & \quad Y^- \rightarrow B^- \\
  s_R : & \quad A^+ \rightarrow B^+, & \quad A^- \rightarrow B^- \\
\end{align*}
\]

\( p \) Parameter learning: fine-tunes the memory elements/parameters \((u,v)\) of the network. Therefore, it is performed after structure learning. Parameter learning can be any supervised learning algorithm. Normally, learning algorithm that seeks to minimize certain cost function \( p(X,Y,u,v) : E_s + E_p \rightarrow \varepsilon \) is employed. \( E_s \) is the structure learning error, and \( E_p \) is the parameter learning error. \( \varepsilon \) is a constant.

In CLFNN, the behavior is represented by a linguistic term characterized by fuzzy set \( A \). \( A \) is a mapping of \( X \) to the linguistic representation of the CLFNN. Each value of \( X \) has \( J \) corresponding \( A \) describing it. In other words, each \( A \) defines a set of locations in the input space. They are characterized by their membership function \( \mu_A \), which is described by weights/memory elements \((u,v)\) (for trapezoidal membership function). An example is given below, where \( \min_i \) and \( \max_i \) refer to minimum and maximum value of \( i \)th dimension input respectively, as given in (2).

\[
\mu_{A_i}(x_i \in X) = \begin{cases} 
1, & u_i \leq x_i \leq v_i \\
\frac{x_i - \min_i}{u_i - \min_i}, & x_i < u_i, x_i \geq \min_i \\
\frac{\max_i - x_i}{v_i - \min_i}, & x_i > v_i, x_i \leq \max_i \\
0, & \text{otherwise} 
\end{cases}
\]  

(2)

The lateral inhibition of complementary learning is implemented through (3) and (4). When \( D^+ \) is presented, \( R^+ \) will be activated, and concurrently, \( R^- \) will be deactivated, and vice versa.

\[
\mu_{R^+}(X) = \begin{cases} 
1, & X \in D^+ \\
0, & \text{otherwise} 
\end{cases}
\]  

(3)

\[
\mu_{R^-}(X) = \begin{cases} 
1, & X \in D^- \\
0, & \text{otherwise} 
\end{cases}
\]  

(4)

This is the functional model of the complementary learning paradigm, which minimizes the confusion in the inference process. Furthermore, modeling of the source space \( <X,Y> \) with \( R^+ \) and \( R^- \) gives a better representation of the problem dynamics.

Fig. 1 shows the CLFNN model. It has five layers; each layer corresponds to the Mamdani’s fuzzy rule model as described in (5).

\[
R : \text{IF } X \text{ is } A, \text{ THEN } Y \text{ is } B
\]  

(5)

These rules are used by the system to infer, and can be used for system validation as well.

B. CLFNN and Imbalanced Data

Since CLFNN constructed its knowledge base for positive and negative classes separately, the influence of each class on the other is minimized. Hence, CLFNN is less susceptible to data imbalance. Furthermore, the inference process of CLFNN makes use of the lateral inhibition, which could potentially improve the system performance in pattern recognition, even when it is imbalanced dataset.

CLFNN inference is based on the difference between the positive and negative. Furthermore, the positive and negative spaces are separated, as illustrated in (6).

\[
y = \begin{cases} 
+, & \text{if } x \in X^+ \text{ AND } \mu_{R^+}(x) > \mu_{R^-}(x) \\
-, & \text{if } x \in X^- \text{ AND } \mu_{R^+}(x) < \mu_{R^-}(x)
\end{cases}
\]  

(6)

As shown, even if the sample of the minority is less, it will not affect the system as much because CLFNN has separated the positive and negative regions. If \( x \) is not in the space covered by the positive and negative rule spaces, it will be inferred based on the distance from the positive and negative rule spaces.

This is in contrast to

1) Positive/Negative System

Positive/Negative system creates its knowledge based only on the target class. They totally ignore the contribution
of another class. One example is the decision tree C4.5. For C4.5 or any decision tree algorithm, because they are recursive partitioning algorithm, minority class is difficult to learn. This is because, as progressing down the tree, the number of samples decreases. This poses a great challenge for the already limited minority class. The matter becomes worse if the pruning method is not set properly, as it will remove the node with minority class.

Another popular tool from positive/negative系统 is SVM. SVM yields poor result when it comes to imbalanced dataset [9]. The unsatisfactory performance is seen by the way the systems make decision, \( y \), as given in (7).

\[
y = \begin{cases} +, & \text{if } \text{sgn} \left( f(x) = \sum_{i=1}^{T} y_i \alpha_i^t K(x, x_i^r) + b \right) = 1 \\ -, & \text{if } \text{sgn} \left( f(x) = \sum_{i=1}^{T} y_i \alpha_i^t K(x, x_i^r) + b \right) = -1 
\end{cases}
\]

(7)

where \( \alpha' \) is the parameter obtained by solving the primal Langrangian, \( b \) is the bias, and \( K(x, x_i^r) \) is the kernel function. These are the three parameters that delineate a SVM system. The \( \alpha' \) of the minority class tends to become much larger than those of majority class because of the low presence of minority class instances makes them appear to be farther from the hyperplane. Thus, SVM is greatly affected by the data imbalance. Some have proposed new kernel function to ameliorate the imbalanced dataset problem [1][9], albeit the most popular is still the original SVM.

2) Neutral Learning

Neutral learning does not consider positive or negative. The system merely optimize a given accuracy measure or cost function. Owing to this reason, MLP suffers in poor performance for imbalanced dataset because the accuracy used does not reflect the system performance accurately due to the data imbalance.

\[
y = \begin{cases} +, & \text{if } f \left( \sum_{j=1}^{I} w_j \left( f_j \left( \sum_{i=1}^{I} w_{ij} x_i + \text{bias}_j \right) \right) \right) > \rho \\ -, & \text{Otherwise.} 
\end{cases}
\]

(8)

where \( f_j \) = activation function of hidden neuron, and \( f_k \) = activation function of output neuron; \( w_{ij} \) = weight of the connection between the \( j^\text{th} \) input neuron and the \( i^\text{th} \) hidden neuron. Likewise, \( w_{ij} \) = weight of the connection between the \( j^\text{th} \) hidden neuron and the \( k^\text{th} \) output neuron. Since MLP adjusts the weights according to the error signal at the output layer, the weights of the minority class is tuned relatively lesser than the majority class. Hence, the weights will be biased to the majority class, and as a result, the minority class will seldom able to pass the threshold test.

Another example of neutral learning is LDA. Since LDA classification requires the computation of covariance matrices (see (9)), LDA is severely affected by data imbalance since the prior probability of the majority class overshadows the differences in the sample covariance matrix terms [4]. LDA will perform well for imbalance data if and only if the sample covariance matrices are identical. Oftentimes, this is not true in real world application however.

\[
S_m = \sum_{i=1}^{T} \left( x_i^r - \bar{\mu}_m \right) \left( x_i^r - \bar{\mu}_m \right)^T
\]

(9)

where \( S_m \) is the covariance matrix of class \( m \), \( T_m \) is the number of samples belonging to class \( m \), and \( \bar{\mu}_m \) is the mean for class \( m \).

Thus, complementary learning system demonstrates a superior structure for imbalanced dataset. The segregation of positive and negative knowledge lessen the effect of imbalanced data on the system learning and inference.

C. Experimental Setup

Four datasets are used for assessing CLFNN performance in imbalanced dataset. The datasets are shown in Table 1. The imbalance ratio is calculated by dividing the positive by the negative. In medical field, positive refers to the cases of having the disease, and negative refers to normal cases.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Number of positive</th>
<th>Number of negative</th>
<th>Imbalance Ratio</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPECT</td>
<td>212</td>
<td>55</td>
<td>3.85</td>
<td>[16]</td>
</tr>
<tr>
<td>WBBCD</td>
<td>212</td>
<td>357</td>
<td>1.65</td>
<td>[16]</td>
</tr>
<tr>
<td>FNA</td>
<td>235</td>
<td>457</td>
<td>0.51</td>
<td>[17]</td>
</tr>
<tr>
<td>Thermogram</td>
<td>5</td>
<td>65</td>
<td>0.07</td>
<td>[18]</td>
</tr>
</tbody>
</table>

Abbreviation: SPECT- Single Photon Emission Computed Tomography; WBBCD-Wisconsin Breast Cancer Diagnostic; FNA-Fine Needle Aspiration

As seen in Table 1, all the dataset are imbalanced. The closer the imbalance ratio is to one, the more balance is the dataset. Hence, thermogram dataset is the most extreme imbalanced case. Each of the dataset is divided into three stratified cross-validation sets of training and testing. Training set is made up of 1/3 of the data, whereas testing set is made up of the remaining unseen 2/3. The class distribution is maintained. All simulations are run on the same system, i.e. Pentium 4 2 GHz, with 1 GB RAM. Same training and testing sets are used for every algorithm.

The measure employed is the F-measure, as shown in (10).

\[
F-Measure = \frac{(1 + \beta^2)^{\text{recall} \times \text{precision}}} {\beta^2 \times \text{recall} + \text{precision}}
\]

(10)

where \( \beta \) corresponds to the relative importance of precision versus recall. It is set to one as false alarms (false positive) and misses (false negative) are considered equally costly. Recall and precision are given in (11) and (12).

\[
\text{Recall} = \frac{\text{True Positive}} {\text{True Positive} + \text{False Negative}}
\]

(11)

\[
\text{Precision} = \frac{\text{True Positive}} {\text{True Positive} + \text{False Positive}}
\]

(12)

For illustration purpose, an instance of CLFNN named
FALCON-AART [14] is used to classify the four datasets (The relationship between the 9-tuple model and FALCON-AART is presented in Appendix). The averaged performance over the three cross-validation sets is compared against popular tools MLP, RBF, C4.5, SVM, and LDA.

### III. Experimental Results

The averaged performance of CLFNN and others are summarized in Table 2.

<table>
<thead>
<tr>
<th>Method/Dataset</th>
<th>SPECT</th>
<th>WBCD</th>
<th>FNA</th>
<th>Thermogram</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>0.762</td>
<td>0.925</td>
<td>0.917</td>
<td>0.2</td>
</tr>
<tr>
<td>RBF</td>
<td>0.8</td>
<td>0.912</td>
<td>0.912</td>
<td>0.2</td>
</tr>
<tr>
<td>C4.5</td>
<td>0.763</td>
<td>0.907</td>
<td>0.906</td>
<td>0.2</td>
</tr>
<tr>
<td>SVM</td>
<td>0.79</td>
<td>0.93</td>
<td>0.93</td>
<td>0.2</td>
</tr>
<tr>
<td>LDA</td>
<td>0.90</td>
<td>0.931</td>
<td>0.918</td>
<td>0.2</td>
</tr>
<tr>
<td>CLFNN</td>
<td>0.91</td>
<td>0.96</td>
<td>0.96</td>
<td>0.37</td>
</tr>
</tbody>
</table>

Figures given are the F-measure

As shown in Table 2, CLFNN generally has better performance than other methods. Regardless of the data imbalance ratio, CLFNN outperforms the popular method in terms of F-measure. The higher the F-measure, the greater is the accuracy in classifying the dataset. Thus, CLFNN exhibits the superior performance in balance and imbalanced dataset. In thermogram dataset, an extreme imbalanced example, none of the system can effectively classify it.

In general, the higher the imbalance ratio, the more difficult for the system to learn the problem. Since it is often not known that the training set reflects the real data distribution, it is desirable for the system to perform consistently over different data distribution [2]. In order to investigate the system performance over different data distribution, a graph is plot for F-measure against the imbalance ratio, and is shown in Fig. 2.

![Fig. 2. System performance under different data distributions.](image)

As shown in Fig. 2, CLFNN is less susceptible to the effect of imbalance ratio. It performs relatively consistently when compared against other computational intelligence methods, suggesting its capability of learning imbalanced dataset. Generally, as the imbalance ratio increases, the system performance drops. In this work, CLFNN is illustrated by FALCON-AART, as it uses distance-based clustering in its structure learning and modified backpropagation to tune its parameters, these may cause the system to be affected by the imbalanced data distribution.

### IV. Conclusion

Real world dataset is often imbalanced, especially in medical dataset. Thus, a computational intelligence tool to be applied in these areas should have the ability or robustness to perform well under different data distributions. They should offer a consistent performance even when imbalanced dataset is encountered. Unfortunately, conventional computational intelligence methods could not effectively learn the imbalanced data. Thus, many endeavors have been made to alleviate the effect of imbalanced dataset. Altering the data distribution is the most popular method owing to its simplicity. However, this changes the real data distribution and may affect the system performance in overall. Realizing that, the complementary learning fuzzy neural network is proposed. It is inspired by the positive and negative learning mechanism observed in human brain. Complementary learning segregates the positive and negative knowledge, and then exploits the lateral inhibition between the positive and negative knowledge. Incorporating this into the fuzzy neural network may offer superior recognition performance, and may be less susceptible to the data imbalance. Preliminary theoretical and empirical analyses show that CLFNN gives relatively more consistent performance over different data distributions, suggesting CLFNN as a promising tool for handling imbalanced dataset. In future, a structure and parameter learning that are not affected by imbalanced data can be used to improve the present system. More experiments will be conducted to investigate CLFNN as a tool to imbalanced dataset.

### References


[7] Y. Liu, A. An, and X. Huang, “Boosting prediction accuracy on imbalanced datasets with SVM ensembles”, in W. K. Ng, M.
APPENDIX

The FALCON-AART [14] is an example of the CLFNN model proposed. FALCON-AART has the same structure as the one depicted in Fig. 1. FALCON-AART makes use of a modified Adaptive Resonance Theory (ART) algorithm to learn from positive and negative examples separately. This modified ART constructs the $A$ and $B$ from $X$ and $Y$, respectively. Apart from that, modified ART also constructs the rulebase $R$ to link $A$ and $B$. In other words, modified ART is the structure learning method, $s$, of FALCON-AART. After the structure is formed, adaptive back-propagation learning with momentum is applied to fine-tune the parameter of FALCON-AART. Possessing the 9-tuple, FALCON-AART is shown to be a CLFNN model.