# An Optimized Hybrid Fuzzy Weighted k-Nearest Neighbor with the Presence of Data Imbalance

Soha A. Bahanshal[1], Rebhi S. Baraka[2], Bayong Kim[3], Vaibhav Verdhan[4]
Department of Computer Science, University of Massachusetts Lowell, Lowell, USA[1, 3]
Department of Computer Science, Islamic University of Gaza, P. O. Box 108, Gaza, Palestine[2]
Analytics Leader, AstraZeneca, London, UK[4]

***Abstract*—We present an optimized hybrid fuzzy Weighted k-Nearest Neighbor classification model in the presence of imbalanced data. More attention is placed on data points in the boundary area between two classes. Finding greater results in the general classification of imbalanced data for both the minority and the majority classes. The fuzzy weighted approach assigns large weights to small classes and small weights to large classes. It improves the classification performance for the minority class. Experimental results show a higher average performance than other relevant algorithms, e.g., the variants of kNN with SMOTE such as Weighted kNN alone and Fuzzy kNN alone. The results also signify that the proposed approach makes the overall solution more robust. At the same time, the overall classification performance on the complete dataset is also increased, thereby improving the overall solution.**

***Keywords*—Imbalanced data; fuzzy weighted kNN; SMOTE; classification model; optimized hybrid kNN**

## I. Introduction

In supervised learning, labeled training data is used to prepare certain classifiers and find the class name of the test data using that classifier [1]. The performance of such classifiers on balanced datasets is generally better than on imbalanced datasets. Hence, there is an increasing need to tackle the issue of class imbalance [2, 3]. The problem of class imbalance states that the number of instances in one class is slightly lower in these datasets than in the other classes [4]. On imbalanced datasets of the binary class, only one positive and one negative class is present. The positive and negative classes are the minor and the major classes, respectively.

In many classification problems, however, the more useful are the instances of the minor with lower instances [5]. Therefore, imbalance occurs whenever the class of interest is relatively rare and has a small number of instances compared to the majority class. In addition, relative to the cost of misclassifying the majority class, for example, the cost of misclassifying the minority class is very high; consider cancer versus non-cancer or fraud versus un-fraud [6]. Since the majority class is over-represented, it impacts the training of the classifier and hence the majority class has better accuracy than the minority class(es).

Although a variety of solutions to data imbalance have been developed, in some ways they have shortcomings. Some solutions consider adding, deleting or weighting the data in order to closely balance the data. Other solutions attempt to find some good pre-processing measures for solving such particular problems in the training dataset that may restore balance between the majority and minority classes before performing classification [7, 8].

In general, existing class-imbalanced classification methods are divided into four categories: either manipulating or modifying the distribution of data by under- or over-sampling (data sampling), modifying in traditional classification existing algorithms to suit class imbalance (algorithmic modification), ensemble approaches [9], or cost-sensitive learning [10]. Therefore, data imbalance can be resolved by over-sampling the under-represented class or under-sampling the over-represented class. But both of these methods are not much scientific and suffer from various drawbacks. Therefore, we adopt the first and the second categories of sampling a dataset in the pre-processing phase and the modification of traditional classification algorithm. The data sampling methods focus on balancing the data, and the common strategies are to reduce the majority class examples (under-sampling) or to add new minority class examples to the data (oversampling) [8, 11].

Synthetic Minority Oversampling Technique (SMOTE) is one of the techniques used to balance imbalanced datasets. Researchers have widely adopted SMOTE due to its versatility and added value with respect to random over-sampling [8]. It reduces the possibilities of over-fitting by randomly resampling the data and generating new samples of the minority class by interpolating multiple samples of the minority class that lie together. Nearest neighbour rule can be used over here. The over-fitting dilemma is thereby eliminated and the decision space is more widespread for the minority class; meanwhile the decision space for the majority class is reduced. By operating in feature space, synthetic instances are created. Some drawbacks of SMOTE, however, are unavoidable because it synthesizes new instances without taking the majority class into account [8], which could lead to fuzzy boundaries between the positive and negative classes. Therefore, SMOTE technique has been proposed with various improvements and extensions that aim to eliminate its drawbacks.

In this paper, we combine a method of class weighting with SMOTE's over-sampling of the minority class to improve the classification accuracy of the minority class without sacrificing the accuracy of the majority class. The combination is performed in a simple classification model based on kNN algorithm [12, 13] which has the ability to accommodate enhancements and extensions [14]. It is the Hybrid Fuzzy weighted k Nearest Neighbor (HFWkNN) classifier introduced

in [15]. It is a weighted and fuzzy extension to kNN based on fuzzy set theory.

We optimize HFWkNN to deal with imbalanced datasets and find greater results in the general classification of imbalanced data for both the minority and the majority classes. It determines the fuzzy membership function in favor of the minority class and creates a fuzzy equivalent relationship between the unlabeled instance and its k closest neighbors. In other words, it takes into account the fuzziness of an instance's closest neighbors, which can decrease the disturbance of the majority class to the minority class. The advantages of the neighbor weighted K nearest neighbor method are combined with fuzzy logic, i.e., the assignment of large weights to small classes and small weights to large classes. Fuzzy classification tends to more adequately classify objects as it defines how much of an object belongs to a class.

As presented in [15], the hyperparameters $\gamma$, $\varepsilon$ and $\varepsilon_{min}$ are introduced in the membership function of HFWkNN to increase the accuracy score, improve the performance and handle the class-imbalance. Weight-assignment technique is developed and combined with SMOTE for the class membership function of the HFWkNN of each neighbor, which learns the class weight for each training sample, to process imbalanced data. The minority class samples are given a higher weight to let the classifier concentrate on them.

The rest of the paper is organized as follows. Section II briefly discusses some related work. Section III presents the classification model with the class-imbalanced data. Section IV presents the experimental results, analysis and evaluation of the model. Finally, Section V concludes the paper and suggests future work.

## II. Related Work

There are quite a few innovative solutions and methods which have been proposed by researchers to tackle the problem of class imbalance in classification problems. These methods either oversample the minor class or under-sample the majority class [16]. That is why these methods are sometimes called as sampling techniques. Although these methods are popular, they suffer from the problem of impacting the original distribution of the data. Nevertheless, there are approaches which deal with the issue of class imbalance while not impacting the original structure of the data. such approaches can be utilized by many classifiers such as those based on kNN.

kNN is one of the most utilized and quite popular classification algorithms. It is used in various classification problems and is considered as one of the top 10 algorithms in data mining [14]. But the classic kNN algorithm is not equipped enough if the dataset is imbalanced. Hence, to tackle the issue of imbalanced dataset for kNN algorithm, researchers have proposed quite a few distance or similarity based classification algorithms like kENN [25] and CCW-kNN [17]. But these methods are good for numerical data points.

Weighted kNN proved to performs well on imbalanced datasets. Dubey et al. [18] proposed class based weighted approach for performing classification on imbalanced dataset. In this approach, the distribution of the nearest neighbour was analysed and used to calculate the weights. Classic kNN is used to perform the initial classification and is used to get the respective weights for each of the classes in the classification problem. A hybrid approach was proposed by Patel et al. [19]. It tackles the class imbalance by assigning small weights to the majority class and large weights to the minority class. Tomasev and Mladeni_c [20] explored the hubness effect which is related to kNN in high-dimensional datasets, where minority class instances lead to higher misclassification errors. With low or medium dimensional datasets, majority class instances lead to misclassification.

Fuzzy solutions have been used for dealing with imbalanced dataset problems. However, not much work has been done in this area. Liu et al. [21] proposed a fuzzy kNN approach for unequally distributed dataset. The dataset had strong relationships between attributes, instances and classes. The approach utilized assigning sized memberships, similarity calculations and integration as the main methods. Sometimes, addressing the problem of data leaks in a classification model may result in data imbalance [22]. Ramentol et al. [23] have dealt with imbalanced dataset in a fuzzy-rough ordered weighted average nearest neighbour algorithm for binary classification. Six weight vectors and some indiscernibility relations are used with these weight vectors. Han and Mao [24] proposed an approach which utilizes fuzzy and rough properties of nearest neighbours data. The approach minimized the biasness owing to a membership function resulting in an advantage to the minority class.

In our case, we deal with data imbalance using easy to compute neighbour weighted with fuzzy kNN. We use the fuzzy kNN algorithm [25] to keep some of the nearest neighbours and utilizes their respective distances as key values. These distances are important as they help in finding the respective membership of the data instances into classes. This approach is further enhanced and refined by utilizing weights of different classes which are based on their respective sizes. The proposed solution strives to resolve the class imbalance by finding the membership function of the imbalanced data. This membership function uses the hyperparameters $\gamma$, $\varepsilon$ and $\varepsilon_{min}$ which we introduced in the proposed solution to tune up the classifier as they determine fuzzy membership and therefore lead the classifier to handle data imbalance. The fuzzy membership function was originally proposed by Keller et al. [25]. In our model, in addition to hypermeterizing the membership function, we use SMOTE with weight assignment function to get the membership of instances into all of the respective classes. Next, we present the proposed classification model that deals with the class-imbalanced datasets.

## III. Optimizing Hybrid Fuzzy Weighted KNN with Imbalanced Data

The Optimized HFWkNN for imbalanced data handles the classification problem on the class-imbalanced mixed type datasets. It is an improved version of kNN and it combines fuzzy logic with weights to give more optimal results of prediction. We have presented the Optimized HFWkNN in full detail in [15]. Next, we summarize it for the benefit of the optimization process with the presence of data imbalance.

HFWkNN, as presented in [15], has two stages. In the first stage, the k nearest neighbors of the train set are calculated

against itself. Once the neighbors are calculated, then the class memberships are calculated with the training set using Equation (1).

$$U_c(x_i) = \begin{cases} 0.51 + 0.49 \times \frac{n_c}{k} \times \varepsilon + \gamma & if\ x \in C \\ 0.49 \times \frac{n_c}{k} \times \varepsilon_{min} + \gamma & if\ x \notin C \end{cases} \quad (1)$$

In the membership assignment, we introduce the hyperparameters γ, ε and $\varepsilon_{min}$ in a fuzzy membership to handle the class-imbalanced issue.

In the second stage, for each instance of the test set the k closest in the train set is calculated, based on the values of k. The resulting class is decided using Equation (2) instead of majority voting performed by kNN algorithm.

$$\mathrm{U_c(x_i)} = \frac{\sum_{j=1}^{k}(\mathrm{U_{cj}}) \times 1/\|\mathrm{x_i - x_j}\|\hat{}(2/(m-1))}{\sum_{j=1}^{k} 1/\|\mathrm{x_i - x_j}\|\hat{}(2/(m-1))} \quad (2)$$

The final class is obtained as the class with the greatest combined votes as a result of Equation (3).

$$\mathrm{C(x)} = \mathrm{argmax_i}\big(\mathrm{U_i(x)}\big) \quad (3)$$

To increase the accuracy score, improve the performance and handle the class-imbalance issue in the data, the hyperparameters γ, ε and $\varepsilon_{min}$ are introduced in the membership function of the proposed HFWkNN model [15]. These hyperparameters are optimized using two different methods which are grid search and random search. These two methods are turned into user defined parameterized callable functions to obtain the values of the three hyperparameters. They are based on the optimization process of the class weight parameter to find the weight for each class.

For instance, by using grid search, large weights are assigned to small classes and small weights are assigned to large classes to minimize the bias of the Optimized HFWkNN towards the majority class and avoid minority class. The following is the pseudocode of hyperparameter optimization procedure of HFWkNN using grid search method as we stated it in [27].

**Step 1:** **Initialize** the different parameters γ, ε and $\varepsilon_{min}$, with cv=5

**Step 2:** **Creating** the search space. we input the domain and the algorithm selects the next value for each hyperparameter in an ordered sequence.

**Step 3:** **Generate** a model using grid search. (grid search technique will construct many versions of HFWkNN with all the possible combinations of hyperparameter γ, ε and $\varepsilon_{min}$ values that are defined)

**Step 4:** **Train** the Model.

**Step 5:** **Train phase:** Once the neighbors are calculated, give each neighbor a weight as the inverse distance of its Euclidean distance from that training data, then find memberships of training data into each class using Equation (1).
The parameters **γ, ε** and **$\varepsilon_{min}$** are introduced in the membership function to give it a weight in all classes. This is to minimize the bias of the classifier towards majority class and avoid minority class.

**Step 6:** **Test phase:** Once the neighbors are calculated, the predicted class is decided as shown in Equation (2) to define the degree of membership of x in each class c.
i = 1, 2, 3, ...C
j = 1, 2, 3, …k
C is the number of classes, k is the number of nearest neighbors and m is the parameter of fuzzy strength.

**Step 7:** The final class is obtained as the class with the greatest combined vote.
Classifier assigns x, using Equation (3), as belonging to the class label whose fuzzy membership for $x_i$ is maximum.

**Step 8:** **Calculate** the model accuracy and save the model configuration and accuracy.

**Step 9:** **Check** if stopping criteria is not complete (no. of iterations end) **Updating** parameter values and **Return** back to step 3.

**Step 10:** **Get** and **report** the optimal value of the parameters and position of the model with high accuracy. Output the settings that achieved the highest score in the validation procedure.

To treat class imbalance, we reduce the bias inherent in the learning procedure and increase the sampling weights for the minority class. The weight-assignment technique is introduced for the class membership function of each neighbor in HFWkNN. Therefore, it learns the class weight for each training sample to process imbalanced data. Weights are assigned to the selected samples according to their importance in the data. The minority class samples are given a higher weight to let the classifier concentrate on them. The minority class decision space is expanded to allow HFWkNN to have a higher prediction on unknown samples of minority class. It also avoids the overfitting problem. Furthermore, combining class weighting with over-sampling of the minority class using SMOTE improves the classification accuracy of minority data without sacrificing the accuracy of the majority class. SMOTE is used to pre-process the dataset.

This over sampling, SMOTE + Weighting_assignment, strategy, can tune HFWkNN towards a certain performance measure of interest with only moderate computational overhead. Each observation is weighted based on the class to which it belongs. The effect of minority class observations is increased simply by a larger weight of these instances and vice versa for majority class observations. This is similar to sampling-based approaches. It takes advantage of two efficient techniques: SMOTE as it is used at the pre-processing phase and class weighting assignment which is used to adjust the class distributions of the imbalanced datasets and respectively weight the base classifiers. This proposed strategy (SMOTE with Weight Assignment) is shown in Fig. 1 and is summarized as follows:

*1)* SMOTE as a common general-purpose approach handles data imbalance of the dataset at the pre-processing phase of data processing.

*2)* We get the weights of the class weight parameter. It is based on the optimization process where we use grid search. Grid search assigns large weights to small classes and small weights to large classes. Such an assignment minimizes the bias of the classifier towards majority class and avoid minority class. The assignments of weights in such a manner assures a better classification performance and tackling of the imbalanced dataset.

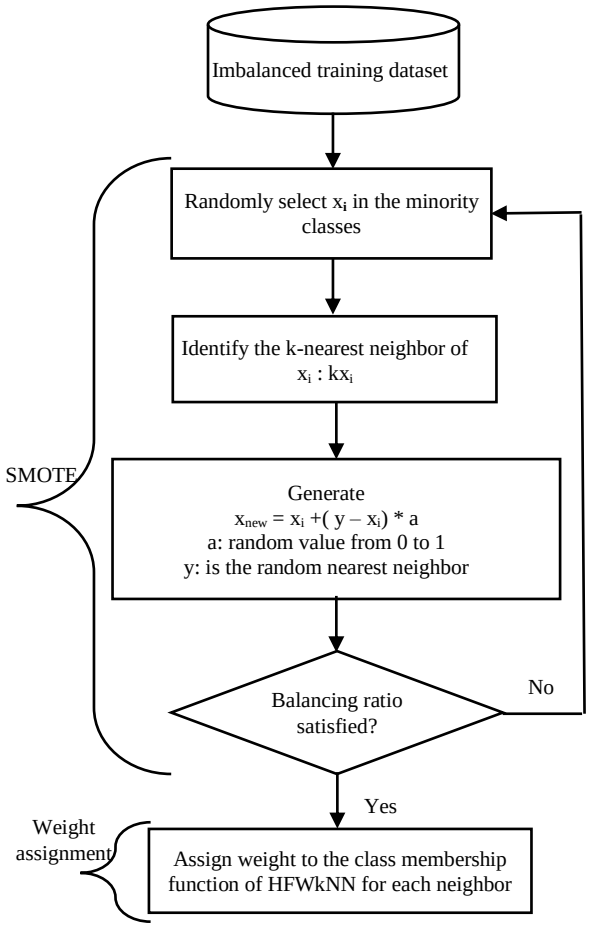


Fig. 1. SMOTE with Weight Assignment.

This strategy employs oversampling, weighting and strengthening. Notice that this improves the minority class samples in boundary region (or uncertain area). It extends the coverage space of minority class samples in boundary region and improves the confidence degree of decision rules without having much impact on the decision space of majority class. Hence the accuracy of HFWkNN is improved. Next, we present the conducted experiments on the optimized HFWkNN based on this strategy and their results.

## IV. Experimental Results

In order to verify the performance of the proposed model on benchmark datasets, we have used five datasets from UCI [26]. These datasets are known to be imbalanced by the unequal distribution of instances into classes. Haberman dataset describes the five year or greater survival of breast cancer patients and mostly contains patients who survive. Pima dataset collected originally from the National Institute of Diabetes and Digestive and Kidney Diseases.

The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset. Ionosphere is a radar data collected by a system in Goose Bay, Labrador. Breast Cancer dataset is a classic and binary classification dataset. Readmission dataset represents 10 years of clinical care for diabetic patients at 130 US hospitals and integrated delivery networks. It includes over 50 features representing patient and hospital outcomes.

Table I provides a summary of these datasets. They vary in the number of instances, number of attributes, class label ratio, and minority class percentage. Based on the minority class percentage, we can see that the datasets are highly imbalanced. For Haberman dataset only 26.5% of patients did not survive. For Pima dataset only 34.9% of patient has diabetes. While for Ionosphere dataset only 35.9% of records labeled as bad radar. For Breast Cancer dataset only 37.3% of patients belong to the malignant class. For Readmission dataset with respect to readmissions only 11% of patients have been readmitted within 30-days.

We compare our approach, the Optimized HFWkNN for imbalanced dataset with kNN, Weighted kNN (WkNN) and Fuzzy kNN (FkNN). We conducted a 5-fold cross validation for each dataset to evaluate the performance of all the 4 algorithms. We obtained and compared the results of Recall and F-value of minority class and the Area Under the ROC Curve (AUC) for each experiment. AUC indicates the overall classification performance and the AUC of a perfect classifier equals to 1, a bad one is less than 0.5. A good classification algorithm usually has a higher AUC.

In all our experiments, we set k to 3 since all the 4 classifiers are based on kNN. The cleaned data was randomized to avoid any selection bias and divided into two parts: 80:20 Training and Test data. This allowed us to train the model on 80% of the data and use an additional 20% to test the performance of the model. A 5-fold cross validation is used to avoid over-fitting on the training data in the evaluated models. The models are trained and evaluated using the same data to ensure fair performance comparison.

Tables II, III, and IV show the results of Recall, F-value of minority class and the Area Under the ROC Curve (AUC) measurements for all of the algorithms with the different datasets. Recall results in Table II indicate that the Optimized HFWkNN outperformed kNN, FkNN. It only outperformed WkNN in the case of the Readmission dataset.

This is due to two factors; first, the improvement occurred in the case of the Readmission dataset because it is larger and richer in terms of attributes than the other 4 datasets. Second, WkNN as it considered weights has shown better performance than kNN and FkNN since they do not consider weights. Nevertheless, the 3 algorithms outperformed the basic kNN since they are fuzzy and weighted.

TABLE I. Short Description of the Datasets

| Datasets | Number of instances | Number of attributes | Class labels (Minority: Majority) | Percent of the minority class |
|---|---|---|---|---|
| **Haberman** | 306 | 3 | 2:1 | 26.5% |
| **Pima** | 768 | 8 | 1:0 | 34.9% |
| **Ionosphere** | 351 | 34 | 1:0 | 35.9% |
| **Breast Cancer** | 570 | 8 | 1:0 | 37.3% |
| **Readmission dataset** | 101,766 | 50 | 1:0 | 11.16% |

TABLE II. RECALL RESULTS FOR THE OPTIMIZED HFWKNN IN COMPARISON WITH OTHER ALGORITHMS

| Dataset | Recall | | | |
|---|---|---|---|---|
| | kNN | FkNN | WkNN | Optimized HFWkNN |
| Haberman | 69% | 73% | 74% | 74% |
| Pima | 72% | 75% | 77% | 77% |
| Ionosphere | 82% | 82% | 83% | 83% |
| Breast Cancer | 87% | 89% | 89% | 89% |
| Readmission dataset | 75% | 76% | 74% | 80% |

The F-value results in Table III indicate that the Optimized HFWkNN outperformed kNN, FkNN and marginally outperformed them in the case of the Ionosphere dataset. This is due to the limited size and number of features in this dataset. Again, it outperformed WkNN in the case of the Readmission dataset due to two factors.

First, the improvement occurred in the case of the Readmission dataset because it is larger and richer in terms of classification-considered features than the other 4 datasets.

Second, WkNN, as it considered weights, has shown better performance than kNN and FkNN since they do not consider weights.

The AUC results in Table IV indicate that the Optimized HFWkNN has a high degree of class separability, i.e., it has high probability of distinguishing between classes. The results also indicate that HFWkNN outperformed kNN, FkNN and WkNN in its ability of class separability especially in the case of Ionosphere and Breast Cancer datasets. This is due to the limited number of classes in these two datasets.

TABLE III. F-VALUE RESULTS FOR THE OPTIMIZED HFWKNN IN COMPARISON WITH OTHER ALGORITHMS

| Dataset | F-value | | | |
|---|---|---|---|---|
| | kNN | FkNN | WkNN | Optimized HFWkNN |
| Haberman | 71% | 76% | 77% | 77% |
| Pima | 72% | 75% | 77% | 77% |
| Ionosphere | 83% | 83% | 84% | 84% |
| Breast Cancer | 87% | 89% | 89% | 90% |
| Readmission dataset | 75% | 76% | 74% | 80% |

TABLE IV. AUC RESULTS FOR THE OPTIMIZED HFWKNN IN COMPARISON WITH OTHER ALGORITHMS

| Dataset | AUC | | | |
|---|---|---|---|---|
| | kNN | FkNN | WkNN | Optimized HFWkNN |
| Haberman | 63% | 67% | 70% | 70% |
| Pima | 67% | 70% | 72% | 72% |
| Ionosphere | 85% | 88% | 88% | 89% |
| Breast Cancer | 87% | 89% | 89% | 90% |
| Readmission dataset | 76% | 76% | 74% | 79% |

The overall results of Recall, F-value and AUC show running the 4 models on the 5 datasets achieved better performance for the proposed Optimized HFWkNN than kNN, WkNN, and FkNN. It increases the classification performance of the minority class compared to the other 3 models. Its performance on the entire datasets is better than the other 3 models. Furthermore, these results indicate that kNN, in all cases has the lowest performance compared to FkNN, WkNN and our Optimized HFWkNN. Therefore, these extensions to kNN are justified and necessary.

It is worth mentioning again that the improvements of the Optimized HFWkNN over the other 3 models are due to combining SMOTE and class weighting assignment. SMOTE has succeeded in improving the accuracy of minority classes. Therefore, the Optimized HFWkNN was able to better model the minority class in the dataset by presenting not only the minority class instances, but also a broader representation of such instances. Such a representation resulted in improving the overall accuracy of the Optimized HFWkNN by concentrating on the minimal cases of the minority, positive classes as well as by properly modelling such classes.

Finally, since the Optimized HFWkNN is based on kNN, its complexity does not differ much from that of kNN which is O(n). Taking the time of computing fuzzy membership grade of training and test samples using Equation (2) above, the time for testing fuzzy membership degree of all classes, and the time of SMOTE (although it is consumed once at the processing stage) and the time of class weighting assignment. These collective times add extra overhead but do not change the overall complexity of the model from that of kNN. Nevertheless, there still a need to perform a complete complexity analysis of HFWkNN.

## V. CONCLUSION

We have presented the Optimized HFWkNN classification model dealing with imbalanced datasets. The model has used three hyperparameters $\gamma$, $\varepsilon$ and $\varepsilon$min that are introduced in the membership function to give it more general character and are tuned to give appropriate membership values for each class and help to balance the dataset. The model also has combined the method of class weighting with over-sampling of the minority class, SMOTE, to improve the classification accuracy of minority class without sacrificing the accuracy of the majority class. This has led to better results in the general classification of imbalanced data for both the minority and the majority.

Experimental results have shown higher average performance for the Optimized HFWkNN than kNN, Fuzzy kNN, and Weighted kNN. Results of Recall, F-value and AUC measurements for the different datasets are higher with the optimized HFWkNN model than the other 3 models. For example, Recall, F-value and AUC measurements with the Readmission dataset are 80%, 80% and 79% respectively, which are higher than those of the other 3 algorithms. These results also prove that the proposed model lead to better overall classification performance on the complete datasets than the other 3 algorithms.

The proposed model can be extended for multiclass and large size datasets with different strategies to construct the

fuzzy membership function in addition to the three hyperparameters $\gamma$, $\varepsilon$ and $\varepsilon_{min}$. Although we have mentioned that the performance of the model is in line with that of kNN, the induced overhead due to computing and ranking fuzzy membership as well as the overhead due SMOTE and class weighting assignment need to be investigated. Finally, the model can be extended to other classification algorithms such SVM based on fuzzy similarities, weights and distances.

## REFERENCES

[1] Kantardzic, M., Data mining: concepts, models, methods, and algorithms. John Wiley & Sons, 2011.

[2] Han, J., J. Pei, and M. Kamber, Data mining: concepts and techniques. Elsevier, 2011.

[3] Patel, H., et al., A review on classification of imbalanced data for wireless sensor networks. 2020. 16(4).

[4] Fernández, A., M.J. del Jesus, and F. Herrera, Hierarchical fuzzy rule based classification systems with genetic rule selection for imbalanced data-sets. 2009. 50(3): p. 561-577.

[5] Kotsiantis, S., et al., Handling imbalanced datasets: A review. GESTS International Transactions on Computer Science and Engineering. 2006. 30(1): p. 25-36.

[6] Abd Elrahman, S.M., A. Abraham, and I. Computing, A review of class imbalance problem. Journal of Network and Innovative Computing. 2013. 1(2013): p. 332-340.

[7] Weiss, G., H. He, and Y. Ma, Foundations of Imbalanced Learning. Imbalanced Learning: Foundations, Algorithms, and Applications. Hoboken. 2013, NJ, USA: John Wiley & Sons.

[8] Chawla, N.V., et al., SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research 2002. 16: p. 321-357.

[9] Barandela, R., et al., Strategies for learning in class imbalance problems. Pattern Recognition. 2003. 36(3): p. 849-851.

[10] López, V., et al., An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information sciences. 2013. 250: p. 113-141.

[11] Estabrooks, A., Jo, T., and Japkowicz, N. A multiple resampling method for learning from imbalanced data sets. Computational intelligence. 2004. 20(1): p. 18-36.

[12] Cover, T. and P. Hart, Nearest neighbor pattern classification. IEEE transactions on information theory. IEEE transactions on information theory. 1967. 13(1): p. 21-27.

[13] Verdhan, V., Supervised Learning for Classification Problems, in Supervised Learning with Python. 2020, Springer. p. 117-190.

[14] Wu, X., et al., Top 10 algorithms in data mining. Knowledge and information systems. 2008. 14(1): p. 1-37.

[15] Bahanshal, S. and Kim B. Hybrid Fuzzy Weighted K-Nearest Neighbor to Predict Hospital Readmission for Diabetic Patients. in 2020 IEEE Symposium Series on Computational Intelligence (SSCI). 2020. IEEE.

[16] García, S. and Herrera, F. Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy. Evolutionary computation. 2009. 17(3): p. 275-306.

[17] Liu, W. and S. Chawla. Class confidence weighted kNN algorithms for imbalanced data sets. in Pacific-Asia conference on knowledge discovery and data mining. 2011. Springer.

[18] Dubey, H. and V. Pudi. Class based weighted k-nearest neighbor over imbalance dataset. in Pacific-Asia conference on knowledge discovery and data mining. 2013. Springer.

[19] Patel, H. and G. Thakur. A hybrid weighted nearest neighbor approach to mine imbalanced data. in Proceedings of the International Conference on Data Science (ICDATA). 2016. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

[20] Tomašev, N. and Mladenić, D. Class imbalance and the curse of minority hubs. Knowledge-Based Systems. 2013. 53: p. 157-172.

[21] Liu, C., L. Cao, and S.Y. Philip. Coupled fuzzy k-nearest neighbors classification of imbalanced non-IID categorical data. in 2014 international joint conference on neural networks (IJCNN). 2014. IEEE.

[22] Rezqa, E. Y. and Baraka, R. S. Document classification based on metadata and keywords extraction," 2021 Palestinian International Conference on Information and Communication Technology (PICICT), 2021. p. 18-24.

[23] Ramentol, E., et al., IFROWANN: imbalanced fuzzy-rough ordered weighted average nearest neighbor classification. IEEE Transactions on Fuzzy Systems. 2014. 23(5): p. 1622-1637.

[24] Han, H. and B. Mao. Fuzzy-rough k-nearest neighbor algorithm for imbalanced data sets learning. in 2010 seventh international conference on fuzzy systems and knowledge discovery. 2010. IEEE.

[25] Keller, J.M., et al., A fuzzy k-nearest neighbor algorithm. IEEE transactions on systems, man, and cybernetics 1985. 4: p. 580-585.

[26] Dua, D. and C. Graff, UCI machine learning repository. School of Information and Computer Science, University of California, Irvine, CA. 2019.

[27] Bahanshal, S. and Kim, B. An optimized hybrid fuzzy weighted k-Nearest Neighbor to predict hospital readmission for diabetic patients. 2021 IEEE 13th International Conference on Computer Research and Development (ICCRD), 2021, pp. 115-120.