ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse108942020.pdf
https://doi.org/10.30534/ijatcse/2020/108942020

A Novel Ensemble Method for Detecting Outliers in Categorical Data

Roy Thomas[1], J.E.Judith[2]
[1] Noorul Islam Centre for Higher Education, Kumaracoil, India, roygptc@gmail.com
[2]Noorul Islam Centre for Higher Education, Kumaracoil,India. judithjegan@gmail.com

ABSTRACT

Outliers are data objects having very important and valuable information, but are rare in their datasets. Several algorithms are developed by various researchers for finding outliers from different types of datasets like multivariate datasets, time series datasets, image datasets and high dimensional datasets. These algorithms are specific to the type of datasets and no general purpose algorithm for detecting outliers in different types of datasets exists in the literature. Moreover most of the algorithms in the literature are capable of dealing with numerical data only. Real world datasets may contain data objects with categorical features in addition to numerical data objects. Here, we propose a novel ensemble learning method for finding outliers in categorical datasets, that ensemble one hot encoder and label encoder together with different outlier detection algorithms such as Local Outlier Factor, One ClassSupport Vector Machine, Elliptic Envelope, Isolation Forest and k-Nearest Neighbor. Experimental results using real world datasets show that the proposed ensemble method for finding outliers in categorical datasets outperforms the other outlier detection techniques.

Key words: Categorical Encoders, Ensemble Learning, Outliers, Unbalanced Dataset.

1. INTRODUCTION

Outlier detection is a research area that extends to various branches of science like statistics, machine learning, data mining, etc. Researchers have proposed various types of algorithms for detecting outliers in different datasets. These algorithms are specific to a particular type of dataset and accept the input data in numerical format only.

1.1 Types of data

The data we encounter in our everyday life can be classified into two types- numeric data and categorical data. Numeric data deal with those data objects whose quantitative nature can be measured using some scales. Categorical data are observed subjectively and deal with qualitative traits that cannot be measured easily. Numerical data is further divided into continuous data and discrete data. Measurements of height, weight, etc. which can have any number of values within a range are examples of continuous data. Discrete data, on the other hand, can have values from a finite set of numbers only. The numbers of students in a class, count of books in a library, etc. aresome examples of discrete data. Categorical data are discrete in nature and can be classified into nominal and ordinal data. A large volume of data objects in the world are nominal data having only a finite set of levels and there exists no relationship or order among these levels. Color is a nominal data which can have a limited set of levels like white, blue, black, etc. There is no order among these levels. Ordinal data are qualitative in nature, but will have an order or sequence among the different values. For example the IQ levels - low, medium, high, etc. possess an order among the values and it is possible to compare these values for certain operations. Binary data is a special type of categorical data which can have only two possible levels. For example the feature ‘gender’, which contains the value either ‘male’ or ‘female’, is a binary nominal data and the feature ‘income level’ which contains the value either low or high is a binary ordinal data.

Similarity metrics are used to measure the similarity among categorical data objects, which is inversely proportional to the distance among them. These similarity measures can be used in data mining and machine learning algorithms that use distance measures for computations. A more common way to deal with categorical features is to use categorical encoders, which convert the categorical values to numeric representation and this corresponding numeric representation is then used in the data mining or machine learning tasks. Different categorical encoders such as one hot encoder, dummy encoder, label encoder, frequency encoder, etc. are using for this purpose. However, they give different results depending upon the problem and the nature of datasets.

1.2 Outliers

Outliers are data objects in a dataset which are very infrequent in the dataset but contain some valuable information. The characteristics of the outliers in a dataset will not be in accordance with the general characteristics of the dataset.

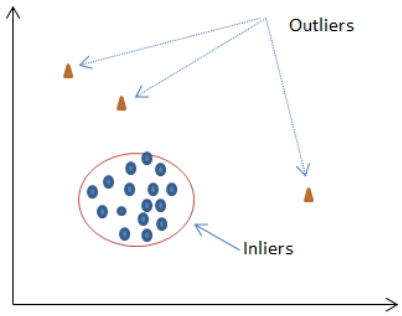

Figure 1: Outliers

The volume of outliers in the dataset is very less compared to the normal objects which form the general behavior of the dataset as shown in Figure 1. The datasets containing the outliers are unbalanced datasets with more than 90% of data objects belong to the normal classes and a very small percentage of data objects belong to the outlier class. Outlier detection is an interesting field of study in different branches of science like statistics, data mining, machine learning, etc. These methods incorporate the application of some specialized techniques to distinguish the rare class from the majority class. Different types of algorithms have been developed to find outliers based on the nature of the data. Outlier detection has a large number of applications like fraud detection in credit card transactions, fault identification in industrial products, medical diagnosis, image processing, malware detection in computer programs, intruder detection, etc.

Outliers are generally classified into three types. They are global or point outliers, conditional or contextual outliers and collective outliers. Global outliers or point outliers are data objects having characteristics different from the characteristics of the remaining data objects in that set. If the difference in characteristics of a minor set of data objects from the major set is based on some condition or context, then these data objects are called conditional or contextual outliers. Collective outliers are data objects as a collection or group forms a rare instance, but individually they are not outliers. For example, consider a class in a school with 40 students in which the normal age of a student in the class is between 12 and 16. If there is a student in that class whose age is 30, that forms a point outlier. If some students come to the class after the class is over, then it becomes a contextual outlier. If one or two students are absent in the class is not an outlier. However, if the majority of the students are absent, itbecomes a collective outlier.

1.3 Ensemble Learning

An ensemble method is the combination of several single methods used to improve the performance in comparison to the individual methods. Ensemble methods are widely used in data mining and machine learning techniques for various application areas. In this paper, an ensemble method is proposed to improve the performance of existing outlier detection models. The basic concept of ensemble learning in outlier detection is to use several outlier detection models for the dataset and then aggregate their predictions to generate a new prediction which outperforms the individual predictions. The effectiveness of ensemble learning depends on the way in which the individual models are constructed and the method used to combine the results.

Voting based and averaging based methods are the two most common ensemble methods used in data mining and machine learning. The base models for the ensemble learning can be constructed by using the same algorithm with different splits of the same training dataset or by using different algorithms with the same dataset. Voting based ensemble methods are mainly used when the predictions are labels and they are further classified into majority voting and weighted voting.

In majority voting, every base model makes a prediction and the most predicted value is taken as the final prediction. In this approach, each model has the same weight. In weighted voting, the number of votes for each model is not the same. Each model is assigned with different weights depending on their importance. Averaging methods are used when the predictions are scores rather than labels and they use the average score of the individual predictions as the final prediction. The averaging method can also be divided into weighted averaging and simple averaging depending on whether or not weights are assigned to different base models.

2. RELATED WORKS

Detecting outliers is an area of research that span in different branches of study like data mining, statistics, machine learning, etc.Hodge and Austin[1] proposed the methods for detecting outliers as emerged from different fields of science and classified the methods into three types. Thomas and Judith[2] accumulated the similarity measures for detecting outliers from categorical datasets and proposed a novel similarity measure for the categorical features based on correlation. Dang, Ngan and Liu [3] proposed a distance based k- nearest neighbor (kNN) algorithm to detect global outliers. Breunig, Kriegel, Ng, and Sander [4] proposed a method to find local outliers. Different classification algorithms for unbalanced datasets were developed by researchers to improve the performance of data mining models. Liu, Ting, and Zhou [5] proposed the decision tree based model ‘isolation forest’and Rousseeuw, Peter, Driessen,and Katrien[6] proposed the elliptic envelope model for classification based outlier detection. The support vector model proposed by Schölkopf, Platt, Taylor, Smola, and Williamson[7] was also used for classification based outlier detection. The algorithms for detecting outliers proposed by the researchers were specific to a particular type of datasets. Hybrid methods as well as ensemble methods were developed for improving the performance of basic outlier detection models. Thomas and Judith[8] proposed a voting based ensemble of three basic outlier detection models to give better performance. Ramesh, Rao and Murthy[9] proposed an adaptive learning scheme ,and Dudi

and Rajesh[10] proposed a plant recognition scheme based on convolutional neural networks.

3. PROPOSED ENSEMBLE METHOD

This paper proposes a novel ensemble method that uses various individual outlier detectors and two encoding techniques. The individual models used in this proposed method have their own advantages and disadvantages. The ensemble model is proposed to minimize the limitations of the individual detectors. The categorical data are first converted into numerical data using one hot encoder and binary encoder separately. The converted outputs from theseencoders aregiven as input to different outlier detection

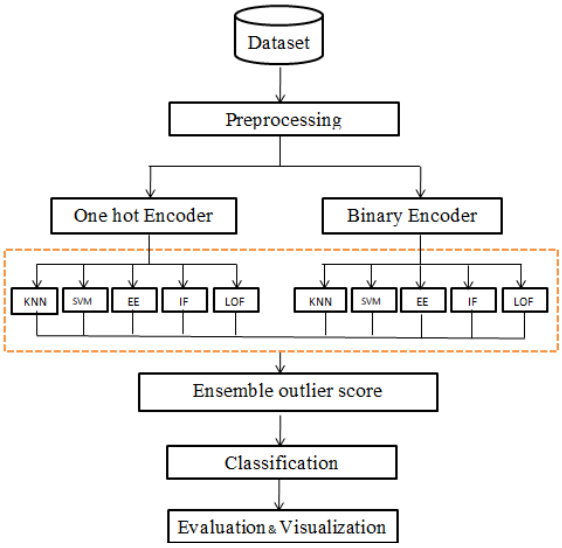

Figure 2:Proposed ensemble method

models. Five dissimilar outlier detection models are used in this method for ensemble learning. The individual outlier detection models used in this method are k-Nearest Neighbor (kNN), Local Outlier Factor (LOF), Elliptic Envelope (EE), Isolation Forest (IF), and One Class Support Vector Machine (OC-SVM). The detailed architecture of the model is shown in Figure 2. The following subsections explain the different steps in the proposed model.

3.1 Preprocessing

Preprocessing is the first step in the proposed method for finding the categorical outliers. The original dataset may contain tuples with missing values. Normally the missing values are comparatively less and the scheme to handle them is to replace with some statistical values derived from the other available values for the particular feature. Here, the model aims to find the outliers, which are less in the dataset. Replacing missing values with such statistical values may severely affect the performance of the detectors. Hence, for our experiment, we avoided the tuples containing the missing values. The datasets may also contain some features, like 'id', which are not relevant for the classification. Such features are also removed from the dataset for the experiment. The next step is to convert the categorical data objects to numerical format. One hot encoder and binary encoder are used for this purpose.

3.2 One Hot Encoder

One hot encoder creates a new feature variable corresponding to each value of the categorical variable. The new feature variable gets a value of 1, if the categorical variable of that data object contains the value of the new variable and all other feature variables get the value 0. One hot encoding is a simple encoding technique and is suitable

Grade
average
excellent
average
good
excellent
average
excellent

Grade_average	Grade_excellent	Grade_good
1	0	0
0	1	0
1	0	0
0	0	1
0	1	0
1	0	0
0	1	0

Figure 3: One hot encoder

Grade
average
excellent
average
good
excellent
average
excellent

Grade_label_encoded
1
2
1
3
2
1
2

Grade_1	Grade_2
0	1
1	0
0	1
1	1
1	0
0	1
1	0

Figure 4: Binary encoder

for categorical encoding when the number of unique values in the categorical variable is comparatively less. If theunique values for the categorical variable are high, then one hot encoder creates many new variables equal to the number of these values and can create serious memory problems. The conversion of the categorical values for the variable 'Grade' using one hot encoder is shown in Figure 3.

3.3 Binary Encoder

Binary encoder is derived from label encoder. Label encoder converts each level for the categorical variable to an integer value. It creates only a single numerical variable corresponding to a categorical variable in which each level corresponds to an integer. Normally, the first level is given a value of 1, the second level a value of 2 and so on. Binary encoder then converts these integer values to their corresponding binary number. The number of bits in the converted number corresponds to the number of new feature variables and the values corresponding to each bit position is the value of that feature. This conversion is shown in Figure 4.

3.4 k- Nearest Neighbor

One of the commonly used proximity based models for finding outliers is k- nearest neighbor(kNN). This outlier detection algorithm is grounded on the assumption that the characteristics of normal data objects are almost similar and they are close together. Outliers are data objects which are far away from the remaining data objects in the data set. In this method, the distance of each data object to its k^{th} nearest neighbor is used as the outlier score of the data object. Normal data objects form dense clusters and the kNN value will be very small compared to the outliers which form sparse clusters or are not part of any clusters. Outlier detection using kNN is computationally expensive as the method calculates the k^{th} nearest neighbor for every data object in the dataset. The optimum choice of the value ofk, which depends on the distribution of the outliers in the data set and the choice of the distance metric, are crucial for the effectiveness of this algorithm. There are different variants of this neighborhood based algorithm such as computing the distance to the k^{th} nearest neighbor, sum or average of the distances to the k nearest neighbors, fuzzy kNN, hybrid kNN, etc.

3.5 Local Outlier Factor

Local Outlier Factor (LOF) is another outlier detection algorithm which computes the outlier score with respect to its neighborhood data objects rather than with all other data objects in the datasets. It is an efficient outlier detection algorithm to find outliers from the features of the data objects in the neighborhood only. The LOF of a data object indicates the density of this data object compared to the density of its neighborhood data objects. Data object whose density is much smaller than the densities of its neighbors belongs to sparse regions and hence is an outlier. The efficiency of the algorithm depends on selecting a suitable parameter value for the number of neighbors to be considered for finding the local density. A small value for this parameter has more local focus and can be mistaken with noise whereas a very high value can miss some local outliers.

3.6 Elliptical Envelope

The elliptical envelope method for determining outliers assumes that a known distribution such as high dimensional Gaussian distribution is formed by the normal objects and tries to find an ellipse in which most of the data objects occur. A data object is considered as a normal object or outlier object subjectto whether or not it occurs inside or outside of the ellipse respectively. This method uses the covariance between the features to estimate the size of the ellipse. This is done by selecting non overlapping subsamples of the original dataset and computing the mean, covariance matrix and the Mahalanobis distance of each data object from the mean for each dimension. This procedure is continued by selecting different subsamples and the distances obtained are used to find the outlier score.

3.7 One class Support Vector Machine

This model is used to find a suitable hyperplane that classifies the data objects into different classes. Here the hyperplanes are used as decision boundaries and data objects belong to either side of the hyperplane form different classes. The objective of this model is to find the hyperplane that has the maximum distance between the classes. Support vectors are data objects that determine the position of the hyperplane. One class-SVM is a classification model that learns the characteristics of one class of data objects only and predicts whether the data objects belong to this class or not. Data objects belong to this class are treated as normal data objects and those data objects which do not belong to this class are classified as outliers.

3.8 Isolation Forest

Isolation Forest is a decision tree based outlier detection model in which the dataset is partitioned into different classes by selecting a random feature and then selecting a random split value for that feature. As the outliers are infrequent and are very much different from the normal data objects, they can be considered as isolated from the most of the data objects. In this approach only a few conditions are needed to separate outliers from the normal data objects. In this model, outliers occur very close to the root of the tree and hence the average path length of outliers is much less than thatof the normal data objects. By using this model, outliers can be found quickly in the decision tree, whereas more partitions are needed to find normal data objects.

3.9 Ensemble method

The proposed ensemble method is a weighted averaging based method which combines the outlier scores obtained from five different base models together with two categorical encoders. Hence a total of ten outputs from individual outlier detection models are used for the ensemble method. The outlier scores obtained from the base models are normalized using min-max normalization to bring all the different outlier scores to a standard scale. The outlier score (P) of the proposed model is calculated from the individual outlier scores obtained from the base models using (1), where $P_i$is the outlier score of the i^{th} model and w_i is the corresponding weight assigned to it.

$$\mathrm{P} = \sum_{i=1}^{10} w_i P_i \qquad (1)$$

3.10 Classification

The outlier score obtained from (1) is used to classify a data object as outlier or inlier. A higher value for this score specifies that the data object is not a normal object and a lower value specifies that the data object is a normal data object. This is determined by a threshold value, which is derived from the hyper parameter ‘outlier fraction’. The

percentage of the outliers in the dataset is given as input. The outlier scores obtained from (1) are sorted and the threshold value is determined using this outlier fraction. Data objects having the outlier score more than the threshold value are considered as outliers.

4. EXPERIMENTS AND RESULTS

The experiments using the proposed model and the results obtained from these experiments are described in this section. The experimental settings and datasets are described in subsection 4.1, andthe results and performance evaluation are described in subsection 4.2.

4.1 Experimental Settings and Datasets

Experiments were conducted in an Intel core i5-based laptop in Python using some of the datasets from the UCI machine learning repository which are publically available. The datasets used for the experiments were the 'census income'dataset[11], 'car evaluation' dataset[12], and 'lymphography' dataset[13]. All these datasets contain categorical features that we used in our different encoder models for converting into numerical features for processing. The full datasets contained many classes. Only two classes were taken for our experiment, as the experiment was to distinguish outliers from the majority of the normal data objects in the dataset. The selection was done randomly in such a way that more than 90% of the data objects belong to a single class and the remaining in another class. The class that contained the minority of the data objects in the unbalanced dataset was taken as the outlier class and was treated as the positive class in the experiment. The characteristics of the datasets used for the experiment are described in Table 1.

4.2 Results and Evaluation

The results obtained from the experiments are tabulated and the values for the evaluation metrics are calculated from these results. The evaluation metrics used for measuring the performance of various outlier detection models are recall, precision, accuracy, F1 score and error rate.In outlier detection models, the class containing the outliers is taken as the positive class and the normal class is taken as the negative class. Table 2 shows the results obtained from the experiment using the census income dataset and the calculated values of these metrics. Table 3 and Table 4 show the results and performance evaluation measures of the car evaluation and lymphography datasets respectively. The precision, recall, F1 score and accuracy of the proposed ensemble method are better than all other individual models. Also the proposed model has the least error rate.

The formulae for computing precision, recall, F1 score, accuracy and error rate are derived from true positive (TP), true negative (TN), false positive (FP) and false negative (FN). Precision (2) is the positive predicted value and recall (3) is the true positive rate. Precision and recall are used to compute the F1-score (4), which is the harmonic mean of precision and recall.

Table 1. Characteristics of Datasets

Dataset	Census Income	Car Evaluation	Lymphograpgy
#instances	121	70	89
#attributes	14	6	8
Attribute type	Categorical, Real	Categorical	Categorical
#classes	2	2	2
#normal	114	65	81
#outlier	7	5	8

Accuracy (5) is the fraction of correct predictions from all the predictions. Error rate (6) is also calculated from all the predictions.

$$Precision = \frac{TP}{(TP+FP)} \quad (2)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (3)$$

$$F1\_Score = 2 * \frac{Precision * Recall}{(Precision + Recall)} \quad (4)$$

$$Accuracy = \frac{(TP+TN)}{(TP+FP+FP+FN)} \quad (5)$$

$$Error\ rate = \frac{(FP+FN)}{(TP+FP+FP+FN)} \quad (6)$$

Table 2. Evaluation result for Census Income dataset

Measure	Precision	Recall	F1-Score	Accuracy	Error Rate
One hot - EE	0.571	0.571	0.571	0.950	0.050
One hot - SVM	0.353	0.857	0.500	0.901	0.099
One hot - IF	0.429	0.429	0.429	0.934	0.066
One hot - LOF	0.600	0.429	0.500	0.950	0.050
One hot - kNN	0.250	0.143	0.182	0.926	0.074
Binary - EE	0.286	0.286	0.286	0.917	0.083
Binary - SVM	0.333	0.286	0.308	0.926	0.074
Binary - IF	0.286	0.286	0.286	0.917	0.083
Binary - LOF	0.429	0.429	0.429	0.934	0.066
Binary - kNN	0.333	0.286	0.308	0.926	0.074
Ensemble	**0.857**	**0.857**	**0.857**	**0.983**	**0.017**

Table 3. Evaluation result for Car Evaluation dataset

Measure	Precision	Recall	F1-Score	Accuracy	Error Rate
One hot - EE	0.500	0.400	0.444	0.929	0.071
One hot - SVM	0.136	0.429	0.207	0.671	0.329
One hot - IF	0.833	1.000	0.909	0.986	0.014

One hot - LOF	0.833	1.000	0.909	0.986	0.014
One hot - kNN	0.750	0.600	0.667	0.957	0.043
Binary - EE	0.200	0.200	0.200	0.886	0.114
Binary - SVM	0.111	0.400	0.174	0.729	0.271
Binary - IF	0.400	0.400	0.400	0.914	0.086
Binary - LOF	0.400	0.400	0.400	0.914	0.086
Binary - kNN	1.000	0.400	0.571	0.957	0.043
Ensemble	**1.000**	**1.000**	**1.000**	**1.000**	**0.000**

Table 4. Evaluation result for Lymphography dataset

Measure	Precision	Recall	F1-Score	Accuracy	Error Rate
One hot - EE	0.714	0.625	0.667	0.944	0.056
One hot - SVM	0.250	0.375	0.300	0.843	0.157
One hot - IF	0.625	0.625	0.625	0.933	0.067
One hot - LOF	0.600	0.375	0.462	0.921	0.079
One hot - kNN	0.667	0.222	0.333	0.910	0.090
Binary - EE	0.500	0.500	0.500	0.910	0.090
Binary - SVM	0.222	0.250	0.235	0.854	0.146
Binary - IF	0.375	0.375	0.375	0.888	0.112
Binary - LOF	0.375	0.375	0.375	0.888	0.112
Binary - kNN	0.667	0.250	0.364	0.921	0.079
Ensemble	**0.750**	**0.750**	**0.750**	**0.955**	**0.045**

The evaluation metrics precision, and recall for the census income dataset are shown graphically using bar charts in Figure 3 and Figure 4 respectively. The graphical representation of F1 score and accuracy are shown in Figure 5 and Figure 6 respectively. The proposed ensemble method for detecting outliers in categorical data has higher values for these metrics, compared to other outlier detecting models. The error rate is shown in Figure 7 as a bar graph in which the proposed ensemble method has the least error rate. From these values, it is clear that the proposed ensemble method for detecting outliers in categorical data outperforms the other methods.

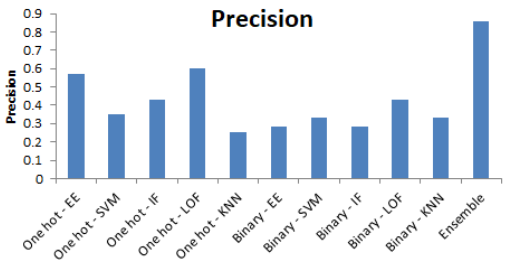

Figure 3: Precision (Census income dataset)

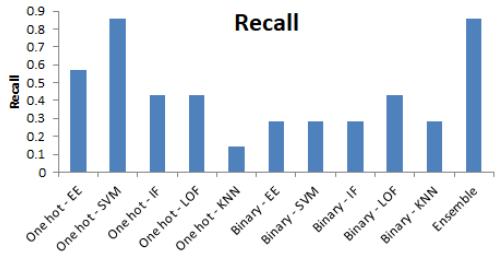

Figure 4: Recall (Census income dataset)

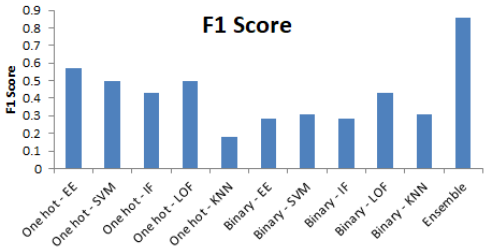

Figure 5: F1 Score (Census income dataset)

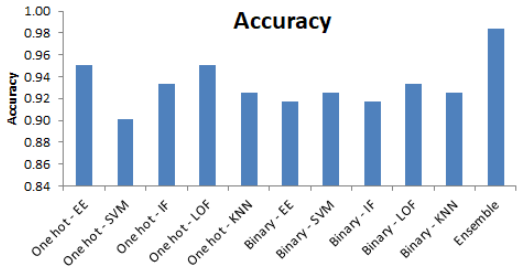

Figure 6: Accuracy (Census income dataset)

Figure 7: Error rate (Census income dataset)

5. CONCLUSION

In this paper, we analyzed the performance of two categorical encoding techniques namely one hot encoder and binary encoder in finding outliers in categorical datasets. These encoders are used to convert the categorical features to numerical data and the converted values are used to detect outliers using different outlier detection algorithms such as Local Outlier Factor, One Class-SVM, Elliptic Envelope, Isolation Forest, and k-Nearest Neighbor. We also proposed a novel outlier detection technique using ensemble method that ensembles two categorical encoders together with these outlier detection algorithms. The performances of these algorithms were evaluated individually and compared their performance with the proposed ensemble method. The evaluation metrics F1-score and accuracy were used for evaluating the performance of the outlier detection methods. Experimental results with real world datasets taken from the UCI machine learning repository showed that the proposed ensemble method for outlier detection in categorical datasets outperforms the individual categorical encoding and outlier detection techniques.

REFERENCES

1. V. J. Hodge, and J. Austin. **A survey of outlier detection methodologies**, *Artificial Intelligence Review*, vol. 22 (2), pp. 85-126, 2004
2. R. Thomas, and J. E. Judith. **Correlation and probability based similarity measure for detecting outliers in categorical Data,***International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol.9 , pp. 2577-2582, January 2020. https://doi.org/10.35940/ijitee.C9053.019320
3. T. T. Dang, H. Y. T. Ngan, and W. Liu, **Distance-based k-nearest neighbors outlier detection method in large-scale traffic data**, *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 507-510, Singapore, 2015.
4. Breunig, Kriegel, Ng, and Sander. **LOF: identifying density-based local outliers**. *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp 93-104,May 2000 . https://doi.org/10.1145/335191.335388
5. F. T. Liu, K. M. Ting, and Z. Zhou. **Isolation forest**, *Eighth IEEE International Conference on Data Mining*, pp. 413-422, Pisa, 2008.
6. Rousseeuw, Peter ,Driessen, Katrien.**A fast algorithm for the minimum covariance determinant estimator,***Technometrics- 41*,pp 212-223. 1999.
7. B. Schölkopf, J. C. Platt, J. S. Taylor, A. J. Smola, and R. C. Williamson. **Estimating the support of a high-dimensional distribution**, *Neural computation* ,Vol.13 pp 1443-1471, 2001
8. R. Thomas, and J. E. Judith.**Voting-Based Ensemble of Unsupervised Outlier Detectors**, *Advances in Communication Systems and Networks*, pp. 501-511. Springer, Singapore, 2020. https://doi.org/10.1007/978-981-15-3992-3_42
9. C. Ramesh, D.V. Rao, and K.S.N. Murthy. **Restoration of Natural Images using Iterative Global and Local Adaptive Learning Scheme**, *International Journal of Advanced Trends in Computer Science and Engineering*,Vol.9 (1), pp 128-133, 2020
10. B. Dudi, and V. Rajesh.**Medicinal Plant Recognition based on CNN and Machine Learning**,*International Journal of Advanced Trends in Computer Science and Engineering*,Vol.8 (4), pp 999-1003, 2019. https://doi.org/10.30534/ijatcse/2019/03842019
11. R. Kohavi, and B Becker.**UCI Machine Learning Repository**, available at http://archive.ics.uci.edu/ml/datasets/adult.
12. M. Bohanec, and B. Zupan.**UCI Machine Learning Repository**, available at http://archive.ics.uci.edu/ml/datasets/Car+Evaluation.
13. I. Kononenko, and B.Cestnik. **UCI Machine Learning Repository**, available at http://archive.ics.uci.edu/ml/datasets/Lymphography.