Journal of Bangladesh Institute of Planners ISSN 2075-9363
Vol. 16, 2023, pp. 19-39, © Bangladesh Institute of Planners

Research Paper

Predicting structure use with machine learning algorithm: Model validation approach for DAP data

A. Biswas[1], S. Roy[1], M. T. A. Shawon[1] and M. M. Rahman[1*]

Abstract

Machine learning techniques have been successfully applied in many fields, including urban planning. The focus of this article is to develop a machine learning model to automatically predict the use of structures. Automatic predictions can help mitigate the heavy load on urban planners in the early stages of decision-making and provide a quick preview of the scenario. In this study, building data from the Detail Area Plan of Dhaka were used. The number of floors and basements in a structure, the structure's age, the number of dwelling units and the structure type were the independent variables for this research. Due to the dataset's inclusion of both numeric and string data, the Decision Tree (DT) classifier was used for prediction. Python routines were used for data cleaning, model development, and model evaluation. The Scikit-learn Python package, primarily used for ML implementation, was utilized to develop the model. The model had an accuracy rate of 91% for predicting the use of institutional, education and research, mixed use, health facilities, under construction, and agriculture structures. Due to incomplete data, residential, restricted and special use, community facilities, miscellaneous, commercial, industrial, transportation and communication use of structures could not be reliably predicted. This model can aid in determining the use of a structure based on the characteristics of the structure (floor, basement, structure type, structure age, dwelling unit), based on historical data for that location. The model demonstrates the use of machine learning in urban planning.

Keywords: Machine learning, structure use prediction, decision tree classifier, DAP, Python.

1. Introduction

As a city expands, it becomes challenging for residents and planners to have a complete understanding of each of its elements like streets and alleys (Lynch, 1960). Urban population and data volume have both risen steeply in the previous half-century (World population by year, 2023). Besides, cities, as multifaceted living laboratories, are increasingly involved in diverse applications to achieve sustainability, resilience, climate adaptation, and managing substantial data in the face of significant environmental and social challenges (Al-Garadi et al., 2020; Bhavsar et al., 2017). This brings us to the concept of 'Big Data,' an essential aspect of smart cities that can significantly contribute to the acquisition of valuable information and facilitate decision-making (Hashem et al., 2016). Simultaneously, the pressures of rapid urbanization and the degradation of quality of life necessitate urban planners to manage growth and implement monitoring strategies, where traditional methods, such as surveys, prove time-consuming and yield

[1] Department of Urban and Regional Planning, Jahangirnagar University, Savar, Dhaka – 1342, Bangladesh.
[*] Corresponding author. Email: mizanurp@juniv.edu

inadequate results (Koutra & Ioakimidis, 2023).

The ever-growing set of computing algorithms known as Machine Learning (ML) (Jordan & Mitchell, 2015; Koutra & Ioakimidis, 2023) can help with this challenge by simulating human intelligence by gathering information from their immediate context (Bell, 2022). Horvitz & Mulligan (2015) state that ML is one of the fastest-growing technical fields at the intersection of computer science, statistics, artificial intelligence, and data science also argued by (Aery & Ram, 2017; Jordan & Mitchell, 2015), and that the use of ML methods has led to more evidence-based decision-making in many fields, including science, technology, commerce, health care, manufacturing, education, financial modelling, policing, marketing, and more (El Naqa & Murphy, 2015). Through repeated learning from training data, ML models recognize patterns and/or minimize the prediction error of complex regression functions (Hagenauer et al., 2019; Robert, 2014). Koutra & Ioakimidis (2023b) assert that the aforementioned methods offer improved evidence-based solutions and decision-making procedures related to urban functioning. Additionally, they successfully address environmental and sustainability problems as well as social issues like integration and urban complexities.

Given the extensive rate of urban expansion and the scale of cities in contemporary times, it has become exceedingly challenging for individuals to effectively manage every facet of a city (Liu et al., 2017). While there is potential for ML to efficiently address urban planning issues (Chaturvedi & de Vries, 2021), the existing literature in this field is limited. Previous research has explored the use of machine learning algorithms for the prediction of criminal activity (Saeed & Abdulmohsin, 2023), land use change prediction (Shen et al., 2020), urban spatial issues often around spatiotemporal topics (Gómez et al., 2019), heterogeneous perceptions of urban space (Ramírez et al., 2021; Zhang, F. et al., 2018). Mahajan et al. (2019), developed a robust prediction model for building age, considering various obsolescence factors, which reduces manual effort and calculation errors. Besides, interpreting remote sensing (RS) datasets has become an increasingly sophisticated and valuable means for comprehending the status and dynamics of both natural and built environments, as modern RS sensors and techniques offer substantial high-quality data with enhanced spatial resolution (Cao et al., 2019). Sun et al. (2021) claimed that ML is a valuable tool for predicting and assessing the performance of building structures, extracting patterns from data collected from various sources, and offering insights for design and assessment, with a focus on its historical development, relevant algorithms, and application areas. ML methods have demonstrated successful applicability in various fields, including the prediction of structural details (Shen et al., 2020).

According to the Institute for Economics & Peace (2022), Dhaka is ranked as the fourth most unsustainable megacity out of the top 20 in the world. The city is characterized by a high population and unplanned urbanization, as highlighted by Rahman et al. (2022). In an effort to address these challenges and promote livability and sustainability, Dhaka has recently implemented the Detailed Area Plan (DAP) for the period of 2016-2035. The DAP is considered the comprehensive master plan for the entire city, aiming to guide its development through proper planning (Rahman et al., 2022; *Rajdhani Unnayan Kartripakkha,* 2023). Although the database in this project includes comprehensive information regarding the structures within its jurisdictional area, it is not without its

shortcomings. These deficiencies primarily stem from human error and various constraints, resulting in incomplete fields and inaccuracies. In addition, new structures within the city are constantly coming up, which are not currently accounted for in the DAP dataset. To achieve effective urban planning, it is necessary to acknowledge and tackle these concerns.

Several studies relevant to building structures, like energy consumption (Pham et al., 2020; Liu et al., 2019), heritage building (Mishra, 2021), earthquake structural safety (Zhang, Y. et al., 2018), have been done for prediction of occupancy and occupant behavior. These studies used different variables like data from temperature sensors and motion and structural occupancy. Common machine learning methods for predicting occupancy and window-opening behavior include logistic regression, ANNs, the Markov chain model, decision trees, *k*-nearest neighbours (KNN), and support vector machines. In this study we used 'Decision Tree Classifier' function for developing the model. Decision trees are often favoured for being easily understood by humans. Providing contextual information and explanations to database administrators or analysts who rely on the model's predictions of structure usage in DAP is important (Sarker, 2021; Mohammed et al., 2016; Jhaveri et al., 2022). To better comprehend the components that contribute to the forecasts, decision trees provide a graphical picture of the deliberation process (Belle and Papantonis, 2021; Krishna et al., 2022). Decision trees lend themselves well to model validation procedures like cross-validation (Myrtveit et al., 2005; Efron, 2004), which are essential to our methodology since they evaluate model performance and guarantee that the prediction model generalizes well to unknown data.

The aim of this article, therefore, is to develop a model to automatically predict the structure use and to evaluate the model's performance. As structure use has various applications in the field of urban planning and development, such as density control, tax collection, and utility services, the model tried to predict the use of the structures. Besides, floor, basement, structure type, structure age, dwelling unit are linked with structure use, for example—a building with no dwelling unit has no possibility to be residential and zero floor indicates that the building is under construction. The use of automated predictions has the potential to solve the significant challenges faced by urban planners in the first phase. This may facilitate the prompt presentation of an overview of the database of DAP for the present situation, specifically pertaining to the utilization of structures.

The subsequent sections of this paper are structured in the following manner: Section 2 of this article delves into the methodology used, encompassing the context, variables, and procedures involved in data cleansing, model selection and training, as well as model evaluation. Section 3 encompasses the results derived from our analysis, while Section 4 comprises the synthesis of findings, accompanied by a thorough discussion and conclusive remarks from the article.

2. Methodology

2.1. Context

The model is suitable for Dhaka, Narayanganj and Gazipur districts as the data set is collected from the attribute table of the structure data of DAP. DAP is the most recent and the largest data set in the field of urban planning in Bangladesh and covers 1,528 km^2

area (RAJUK, 2016), a considerable portion of Dhaka, Narayanganj and Gazipur Districts, with the area of these districts being 1,463.60 km^2 (Dhaka district, 2023), 683.14 km^2 (Narayanganj district, 2023) and 1,770.58 km^2 respectively (Gazipur district, 2023). The structure attributes can display variations based on geographical, social, economic, and environmental factors. It is interesting that Dhaka, Narayanganj, and Gazipur have similar patterns in these properties. Initially, the dataset from the DAP project contained 692,509 records of individual structures.

2.2. Variables

Variables for this study have been chosen based on available data since there has not been much research to forecast the present and future uses of structures. Five independent variables were selected, namely number of floors in a structure (Floor), number of basements in a structure (Basement), type of the structure—*kutcha* (impermanent), *pucca* (masonry or concrete), semi-*pucca* (iron sheets or hybrid), and under construction—(Structure_type), age of structure (Structure_age), and number of dwelling units in the structure (Dwelling_unit). The dependent variable was the use of structure (residential, mixed-use, restricted & special use, community facilities, miscellaneous, commercial, industrial, institutional, under construction, education and research, health facilities, transportation & communication, and agriculture).

2.3. Data cleansing

Data cleansing, generating figures, model development and evaluation were done in this study with routines developed in Python. Microsoft Excel was also used for generating graphs. Data cleansing is a must before analyzing the data by discarding irrelevant information while retaining relevant details (Rizwan & Anderson, 2018). The procedure entails the following actions: first, a comprehensive data check is performed. Second, assigning meaning to the information that was lost. Third, statistical methods are used to analyze the data. Fourth, the data is processed by erasing the null column and axis and assigning 0 to the missing data points. Finally, 'LabelEncoder' function was used from Scikit-learn library to encode the data from the dataset for developing the model. NumPy, Pandas and Matplotlib Python libraries were used for data cleansing purposes.

2.4. Model selection and training

Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, transduction, and learning to learn represent distinct categories within the field of machine learning (Ayodele, 2010). This study adopts Decision Tree (DT) classifier that is under Supervised Machine Learning (SML), where the algorithm creates a function to map inputs to the expected outputs under human supervision (Ayodele, 2010). DT has been used for various purposes in the field of urban planning such as modeling urban patterns connected with the city shape (Cheung et al., 2001), urban structure types (Hecht et al., 2013), urban land cover (Novack et al., 2011), urban growth (Shafizadeh-Moghadam et al., 2017). DT algorithm structures learning data into trees comprising nodes representing attribute tests and leaves representing classes (Bashir et al., 2015; Ruggieri, 2002). DT classifier can handle various types of input data like nominal, numerical and alphabetical (Somvanshi et al., 2016). So, this study uses this method for prediction because the dataset contains both numeric and string data. Besides, the DT

classifier is easy to comprehend and can translate quickly to a set of principles, and previous hypotheses need not be considered to get results (Charbuty & Abdulazeez, 2021).

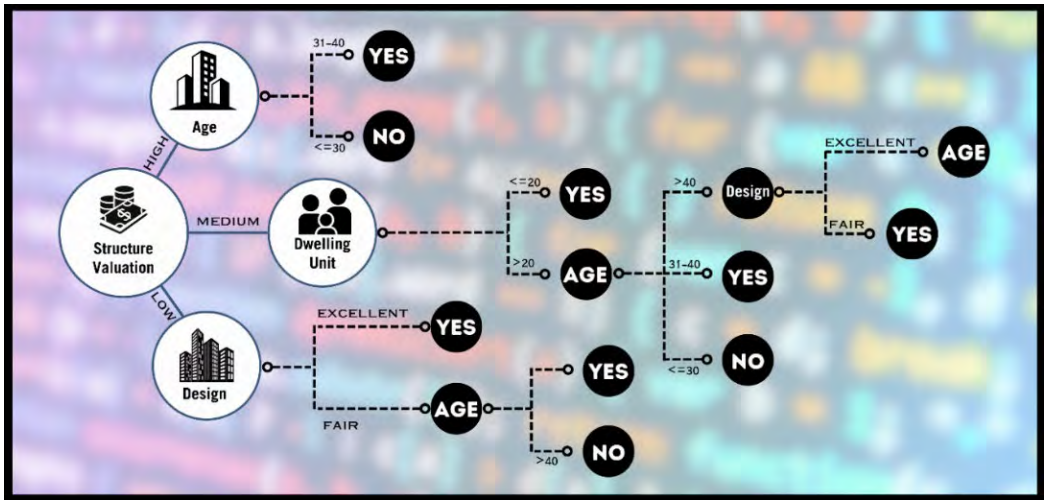

Figure 1. An example of Decision Tree. Source: Adapted from Bengio et al. (2010).

For example, Figure 1 shows a decision tree with respect to buying a house. It has three features: age, dwelling unit, and design. The tree starts with the age feature. If the person's age is less than or equal to 30, the tree splits on the structure valuation feature. If the structure valuation is high, the tree predicts 'YES' (buy the house). If the structure valuation is medium or low, the tree splits on the dwelling unit and design feature. If the design is excellent, the tree predicts 'YES' (buy the house).

The model in this study was developed using the Scikit-learn python library, which is commonly used for implementing ML. The 'train_test_split' function from Scikit-learn was used for creating training and test data. From the dataset, 20% were used for testing the model. The 'DecisionTreeClassifier' function was used for developing the model. In classification, it is common to encounter datasets with an uneven distribution of instances among different class labels. To address this issue, a 'stratified train-test split' method was employed (Brownlee, 2020). This approach ensures that the proportions of examples in each class within the original dataset are maintained when dividing the data into training and testing sets. The parameters for the model are shown in Table 1 below.

Table 1. Parameters of the model.

Parameter	Value
Criterion	Entropy
Random state	42
Max depth	20
Minimum samples leaf	15
Stratify	y

2.5. Model evaluation

From the Scikit-learn library, the 'accuracy_score' function was used for exploring the accuracy of the model, and at the end the 'predict' function from Python was used to predict the structure use for training and testing data.

A learning curve is a graph that illustrates model learning performance over time and is a standard diagnostic tool in ML for algorithms that learn incrementally from a training dataset. Examining learning curves during training can help diagnose learning issues, such as an underfit or overfit model, and whether the training and validation datasets are sufficiently representative (Brownlee, 2019). According to Biswal (2023), overfitting occurs when a model exhibits high accuracy on the training data but performs poorly on new, unseen test data; and underfitting occurs when a model inadequately learns the patterns in the training data, resulting in poor generalization performance on new data. The learning curve has been developed using 'learning_curve' function from Scikit-learn library.

'Classification_report' from Scikit-learn library was also used as the accuracy metric. Accuracy_score is a poor choice of metric to use when the data is not balanced (Branco et al., 2015), as in this study where the overwhelming use of structures was residential. Instead, in such a situation, metrics such as precision, recall, F1-score are more informative. Precision measures the accuracy of positive predictions, indicates how often the model correctly identifies true positives, calculated as

$$Precision = \frac{True\_positives}{True\_positives + False\_positives}. \quad (1)$$

Recall quantifies the fraction of correctly identified positive predictions, assesses how many true positives were correctly predicted, and higher values are desirable. It is computed as

$$Recall = \frac{True\_positives}{True\_positives + False\_negatives}. \quad (2)$$

F1-score offers a balance between precision and recall, making it valuable when these two metrics have opposing values. It is calculated as

$$F1\_score = 2\frac{Recall \times Precision}{Recall + Precision}. \quad (3)$$

Support signifies the number of occurrences of each class in the dataset (Chouinard, 2023). Values of precision and recall lie between 0 to 1, where 0 indicates the lowest accuracy and 1 indicates the highest accuracy (Huilgol, 2023). Confusion matrix is also used to measure the performance of the machine learning classification (Narkhede, 2018). The current model used 'confusion_matrix' from Scikit-learn library to explore the insights between actual and predicted data.

3. Result

3.1. Floors and basements

Figure 2 shows the distribution of buildings in the data set according to the number of floors. There are 130,352 structures in the range of 1 to 10 floors. The rest of the buildings are in the range of 11 to 20 floors (648), and 20 to 30 floors (69). Out of the total 692,509

rows of building data, 561,437 had no data in the floor column. No data cells have a negative impact on the DT model. Still, most of the records that had no data for the number of floors were retained. The number of floors is a very important structure attribute (RAJUK, 2016; Russell & Wong, 1993), so this variable cannot be ignored to predict structure use. As it is difficult to predict the number of floors, filling the no data cells can make the model biased. On the other hand, removing 561,437 rows, which is a large portion of the data, makes the remaining dataset very small. In any case, these records have important data in fields.

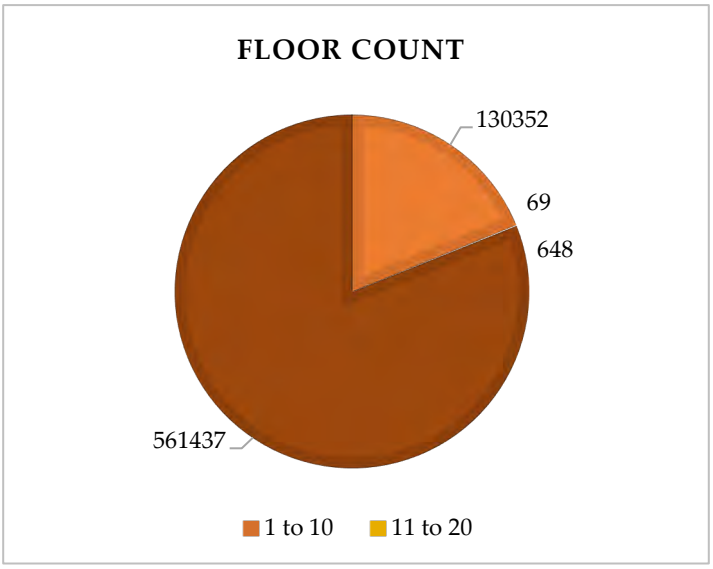

Figure 2. Distribution of buildings by number of floors.

Table 2 shows the number of basements levels in the structures. A total of 692,138 structures have missing data cells. Presumably these buildings have no basement, as this is not common in low-rise, walk-up buildings in Dhaka. Among records with basement data, 340 have a single-story basement, while only 26 structures have basements with 2 to 7 levels. This deficiency in data cells has a detrimental effect on the model. The volume of empty cells is quite large, yet the records were retained. Removing 692,138 rows makes the remaining dataset very small and reduces the performance of the model significantly. Basement is a very important structure attribute (RAJUK, 2016; Russell & Wong, 1993), so this variable cannot be ignored or removed to predict structure use.

Table 2. Frequency of structure by number of basement levels.

Levels of basement	No. of structures
No basement	692,138
1 level	340
2 to 7 levels	26

3.2. Dwelling units

The distribution of structures by number of dwelling units is displayed in Figure 3. There are typically 1 to 10 units per building in the city. The number of such structures is 514,897. The second most frequent class, with 47,333 structures, is between 11 and 20 units. There are just 75 structures of 101 to 300 units. A total of 111,215 records were removed as they could have a negative impact on the model. Some of the records were removed due to inaccurate data, such as solely commercial structures or under construction structures that had dwelling units, and residential structures with zero units. Such incongruent data were possibly due to human errors.

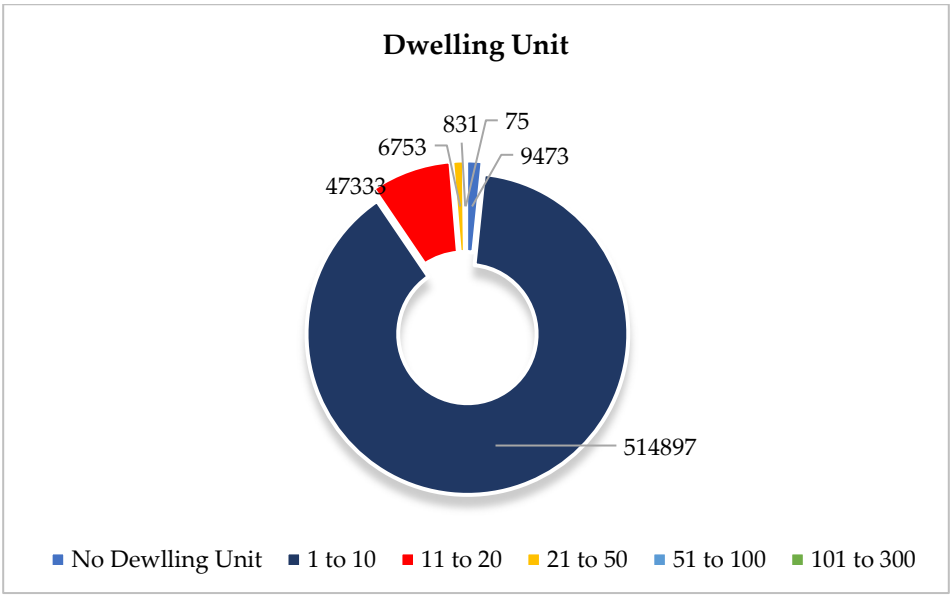

Figure 3. Distribution of structures by number of dwelling units.

3.3. Structure age and type

Figure 4 displays the age of the structures. Structures between 0 and 20 years old are the most prevalent group. The group of structures aged 21 to 30 years comes next. The number of structures older than 30 years is comparatively small. As the model predicts the use of the structures and there are few very old structures, it has the possibility to have a negative impact on the model. Therefore, structures older than 50 years were removed from the database.

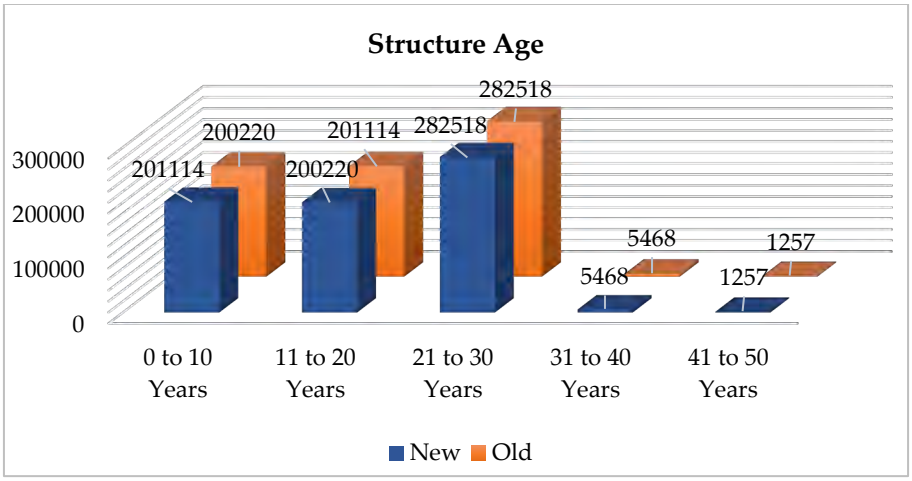

Figure 4. Number of buildings by age of structure before and after data cleansing.

There was no need to remove any records due to data on structure type as there were no missing or erroneous data. The impact of cleaning data for other variable impacted the frequency of structure use data of all use classes were not reduced equally (see Figure 5).

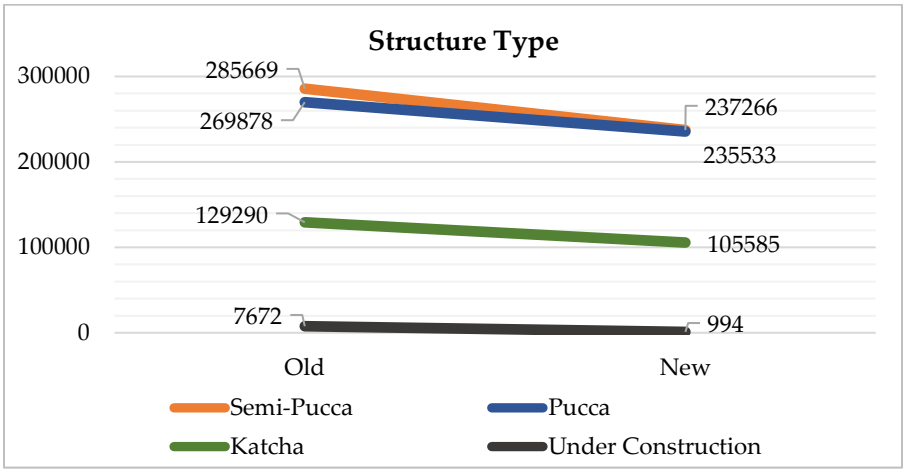

Figure 5. Number of buildings by structure type before and after data cleansing.

3.4. Structure use

Figure 6 shows the number of structures in the DAP database by use, including residential, commercial, mixed-use, etc. The figure displays the old and new numbers of records (before and after removing problematic records). According to the graph, the majority of building are in residential use. Dhaka city has more than 500,000 residential structures which is almost 70% of all structures. Around 7,000 records with residential use were eliminated because of missing data. There were also a lot of records removed in under-construction and commercial use categories because of large numbers of missing data.

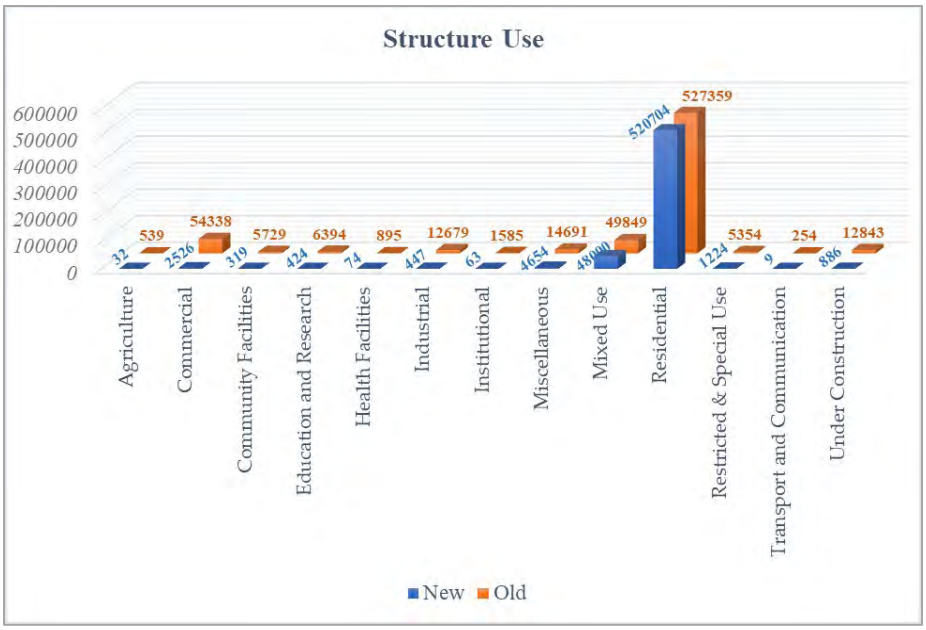

Figure 6. Comparison of number of records before and after cleansing of dataset.

3.5. Training and validation

After cleaning the data, 579,378 records were left for developing the model. Of these, 80% were randomly selected for training the decision tree model, and the rest 20% were used for testing the model. The model provided an accuracy of 91.20 % for the training samples and 91.18 % for the test samples.

As there is an issue of inadequate data for the number of floors and number of basement levels, the model was also run without these variables. The accuracy for the test samples was 91.08% and the accuracy for the training set was 91.06%.

Figure 7 shows that the training and validation scores are sufficiently close to each other. This shows that the variables floor and basement, which include some errors in the dataset, did not affect the acceptance level of the model. The training score refers to the machine learning model's performance on training data during the training phase. The cross-validation score refers to the machine learning model's performance on the testing data, which has some deviation from the testing score in the beginning. The gap continuously decreased as the volume of training data increased, and after getting trained by more than 320,000 records, they overlapped with each other which is an indicator of a good model.

Figure 7. Learning curve of model with number of floors, number of basement levels, structure type, structure age, and number of dwelling unit.

However, learning curve without Floor and Basement showed a major improvement in learning, where training score and cross-validation score directly overlapped each other after getting trained by more than 360,000 records (see Figure 8). That indicates the model performance improves and predicts better with less missing data.

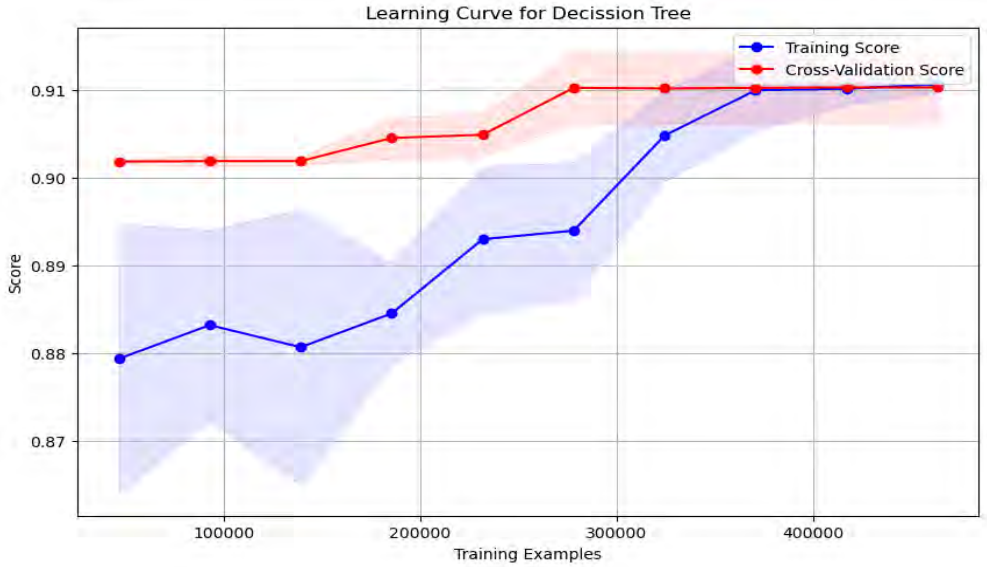

Figure 8. Learning curve of model with structure type, structure age, and number of dwelling units.

3.6. Precision and recall

Precision and recall values were also calculated for the models with five and three variables (see Table 3), and the results were the same for both models.

Table 3. Precision and recall table.

Structure use	Precision	Recall	F1-score	Support
Residential	0	0	0	6
Mixed use	0.62	0.86	0.72	505
Restricted & special use	0	0	0	64
Community facilities	0	0	0	85
Miscellaneous	0	0	0	15
Commercial	0	0	0	89
Industrial	0	0	0	13
Institutional	0.93	0.73	0.81	931
Under construction	0.67	0	0	9,600
Education and research	0.91	1	0.95	104,144
Health facilities	0.61	0.93	0.73	245
Transportation & communication	0	0	0	2
Agriculture	1	0.44	0.61	177
Accuracy			0.91	115,876
Macro average	0.36	0.3	0.29	115,876
Weighted average	0.89	0.91	0.87	115,876

The models have very high precision and recall for some types of structures which means the model is able to predict them with high accuracy (Huilgol, 2023), such as institutional and education and research. The values for mixed use, health facilities, under construction, and agriculture are also good. However, it has low precision and recall for other types of uses, which indicates the model is not able to predict them successfully (Huilgol, 2023) such as residential, restricted & special use, community facilities, miscellaneous, commercial, industrial, transportation & communication.

This variation in accuracy was possibly due to the errors in the data set. A significant portion of data was missing, and records were removed for various reasons, as stated earlier. This may have resulted in zero accuracy for some uses. However, some other uses show very high accuracy because their data were more complete.

Overall, the model has a weighted average precision of 0.89, recall of 0.91, and F1-score of 88 with the accuracy of 91%. This indicates that the model is generally good at predicting some uses, but there is room for improvement.

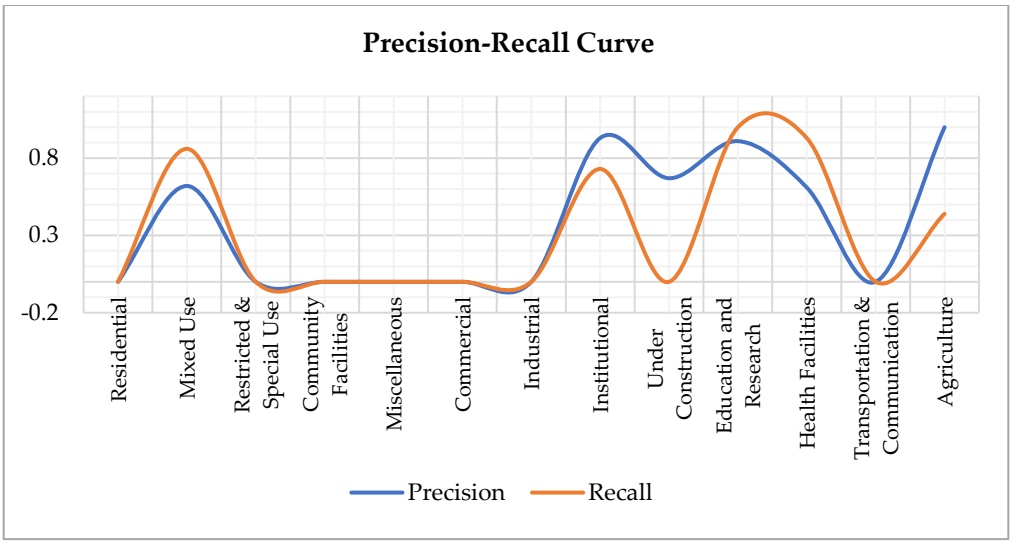

Figure 9. Precision and recall curve.

For most of the types of structures, precision and recall curves are close to each other, the exception being under construction (Figure 9). A slight difference between precision and recall is noticeable for mixed use, institutional, health facilities and agricultural uses.

3.7. Confusion matrix

Figures 10 and 11 show the relation between the actual and predicted use of structures, where education and research, institution, mixed use, heath facilities perform well. In Figure 10, floor, basement, structure age, dwelling unit, structure type are independent variables to predict structure use, whereas, in Figure 11, structure age, dwelling unit, structure type are the independent variables. The two figures display similar results.

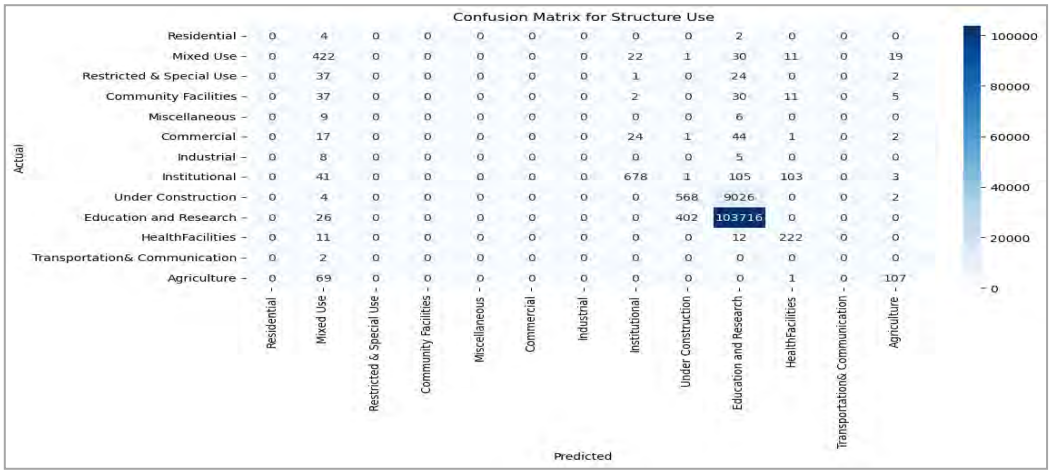

Figure 10. Confusion matrix of structure use (five-variable model).

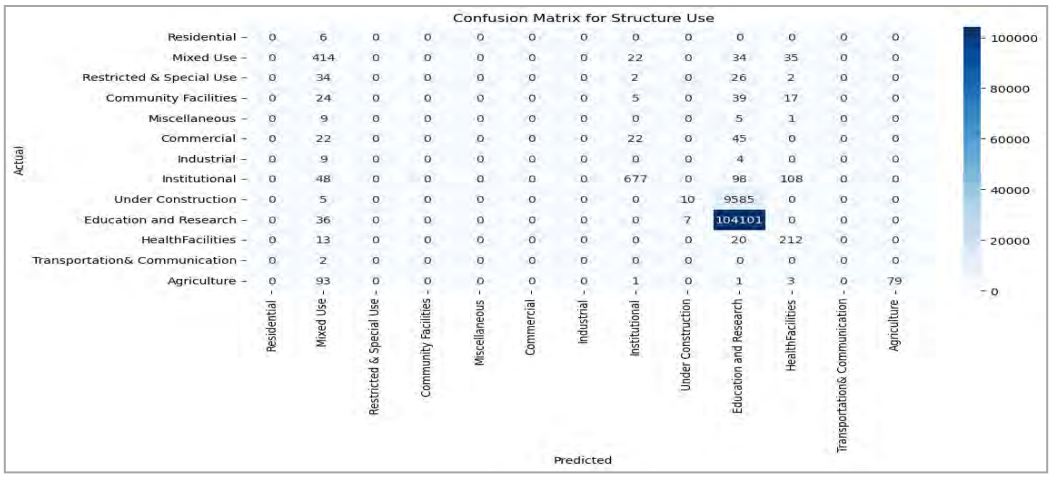

Figure 11. Confusion matrix of structure use (three-variable model).

4. Discussion and conclusion

The study developed a model to automatically predict the use of structures and evaluated the model's performance that can help to release the heavy load on urban planners in the early stage of decision-making and quickly provide a preview of the scenario. The model demonstrates a high level of accuracy in predicting the use, achieving an accuracy rate of 91.20% for the training samples and 91.18% for the test samples, measured through accuracy metric including five variables (floor, basement, structure type, structure age and dwelling unit). The model also provides the same accuracy with three variables (structure type, structure age and dwelling unit) after removing data for floor and basement considering the frequent instances of missing data. The accuracy for the test samples was 91.08% and accuracy for the training set was 91.06%. However, Branco et al. (2015) claimed that accuracy is a very poor choice of metric to use with skewed data. So, precision, recall and F1-score were also calculated for each type of structure use to get more insights. Due to limitation of data the model failed to perform well in predicting some types of use such as residential, restricted & special use, community facilities, miscellaneous, commercial, industrial, transportation & communication. But the model was able to predict institutional and education and research, mixed use, health facilities, under construction, agriculture structure use with 91% accuracy. The confusion matrix also shows that education and research, institution, mixed use and health facilities are addressed better than other types of structures. Furthermore, the training score and cross validation score curves exhibit a significant overlap, indicating that the model effectively learns from a substantial volume of data. This model may help to figure out the use of structures from their attributes (Floor, Basement, Structure_type, Structure_age, Dwelling_unit) based on previous data of a region. But the model learns better when Floor and Basement data are totally removed, which indicates more accurate data can increase the performance of the model.

This DT model can be applied in other regions based on available data. DT classifier can handle input data like nominal, numerical and alphabetical (Somvanshi et al., 2016). Thus, it is suitable for predicting use of structures as structure attributes may include

nominal data.

The practical importance of our results has substantial relevance in real-world contexts. The predictive models developed in this study have the potential for use in several areas and disciplines including prediction of structure usage in the absence of field surveys, enhanced decision-making in planning using remote sensing data, and the optimization of the planning process. Building polygon delineation, structure change detection, structure type classification (Li et al., 2022), and height estimation (Li et al., 2022; Liasis & Stavrou, 2016; Raju et al., 2014) are possible with high quality satellite image, but not the use of structures. However, the process to determine building attributes from satellite image is very time consuming. Besides, the present model is very easy to use and produces easy-to-comprehend results. However, the effectiveness of the model is subject to the quality of the data. The model demonstrates a potential application of ML and its promise for other possible applications.

5. References

Aery, M. K., & Ram, C. (2017). A review of machine learning: Trend and future prospects. *Research Cell: An International Journal of Engineering Sciences*, *25*(63019), 89–96.

Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Communications Surveys and Tutorials*, *22*(3), 1646–1685. https://doi.org/10.1109/COMST.2020.2988293

Ayodele, T. O. (2010). Types of machine learning algorithms. In Y. Zhang (Ed.), *New advances in machine learning*. IntechOpen. doi 10.5772/9385

Bashir, S., Qamar, U., Khan, F. H., & Javed, M. Y. (2015). An efficient rule-based classification of diabetes using ID3, C4.5, & CART ensembles. *Proceedings - 12th International Conference on Frontiers of Information Technology, FIT 2014*, 226–231. https://doi.org/10.1109/FIT.2014.50

Bell, J. (2022). What is machine learning? In S. Carta (Ed.), *Machine learning and the city: Applications in architecture and urban design* (pp. 207–216). Wiley Blackwell. doi 10.1002/9781119815075

Belle, V., & Papantonis, I. (2021). Principles and practice of explainable machine learning. *Frontiers in Big Data*, *4*(688969), 1–25. https://doi.org/10.3389/fdata.2021.688969

Bengio, Y., Delalleau, O., & Simard, C. (2010). Decision trees do not generalize to new variations. *Computational Intelligence*, *26*(4), 449–467. https://doi.org/10.1111/j.1467-8640.2010.00366.x

Bhavsar, P., Safro, I., Bouaynaya, N., Polikar, R., & Dera, D. (2017). Machine learning in transportation data analytics. In M. Chowdhury, A. Apon, & K. Dey (Eds.), *Data analytics for intelligent transportation systems* (pp. 283–307). doi 10.1016/B978-0-12-809715-1.00012-2

Biswal, A. (2023, February 20). *The complete guide on overfitting and underfitting in machine learning*. Simplilearn. https://www.simplilearn.com/tutorials/machine-learning-tutorial/overfitting-and-underfitting#what_is_overfitting

Branco, P., Torgo, L., & Ribeiro, R. (2015). *A survey of predictive modelling under imbalanced distributions*. http://arxiv.org/abs/1505.01658

Brownlee, J. (2019, August 6). *How to use learning curves to diagnose machine learning model performance.* Machine Learning Mastery. https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

Brownlee, J. (2020, June 26). *Train-test split for evaluating machine learning algorithms*. Machine Learning Mastery. https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning-algorithms/

Cao, C., Dragićević, S., & Li, S. (2019). Land-use change detection with convolutional neural network methods. *Environments, 6*(2), 25. https://doi.org/10.3390/ENVIRONMENTS6020025

Chan, J. C.-W., Chan, K.-P. & Yeh, A. G.-O. (2001). Detecting the nature of change in an urban environment: A comparison of machine learning algorithms. *Photogrammetric Engineering and Remote Sensing*, *67*(2), 213-225.

Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, *2*(1), 20–28. https://doi.org/10.38094/jastt20165

Chaturvedi, V., & de Vries, W. T. (2021). Machine learning algorithms for urban land use planning: A review. *Urban Science, 5*(3), 68. https://doi.org/10.3390/urbansci5030068

Chouinard, J.-C. (2023, September 25). *How to use Classification Report in Scikit-learn (Python)*. JC Chouinard. https://www.jcchouinard.com/classification-report-in-scikit-learn/

Dhaka district. (2023, November 9). Bangladesh national information portal. Retrieved November 9, 2023, from https://www.dhaka.gov.bd/bn

Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation. Journal of the American Statistical Association, 99(467), 619-632.

El Naqa, I., & Murphy, M. J. (2015). What Is Machine Learning? *Machine Learning in Radiation Oncology*, 3–11. https://doi.org/10.1007/978-3-319-18305-3_1

Gazipur district. (2023, November 9). Bangladesh national information portal. Retrieved November 9, 2023, from https://www.gazipur.gov.bd/bn

Gómez, J. A., Patiño, J. E., Duque, J. C., & Passos, S. (2019). Spatiotemporal modeling of urban growth using machine learning. *Remote Sensing, 12*(1), 109. https://doi.org/10.3390/RS12010109

Hagenauer, J., Omrani, H., & Helbich, M. (2019). Assessing the performance of 38 machine learning models: The case of land consumption rates in Bavaria, Germany. *International Journal of Geographical Information Science*, *33*(7), 1399–1419. https://doi.org/10.1080/13658816.2019.1579333

Hecht, R., Herold, H., Meinel, G., & Buchroithner, M. (2013). Automatic derivation of urban structure types from topographic maps by means of image analysis and machine learning. *Semantic Scholar*, 6347487.

Horvitz, E., & Mulligan, D. (2015). Data, privacy, and the greater good. *Science*, *349*(6245), 253–255. https://doi.org/10.1126/SCIENCE.AAC4520

Huilgol, P. (2023, July 7). *Precision and recall | Essential metrics for machine learning*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/09/precision-recall-machine-learning/

Institute for Economics & Peace. (2022). *Quantifying peace and its benefits*. http://visionofhumanity.org/resources

Jhaveri, R. H., Revathi, A., Ramana, K., Raut, R., & Dhanaraj, R. K. (2022). A review on machine learning strategies for real-world engineering applications. *Mobile Information Systems*, *2022*, 1833507. https://doi.org/10.1155/2022/1833507

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science, 349*(6245), 255–260. https://doi.org/10.1126/science.aaa8415

Koutra, S., & Ioakimidis, C. S. (2023). Unveiling the potential of machine learning applications in urban planning challenges. *Land, 12*(1), 83. https://doi.org/10.3390/land12010083

Krishna, S., Han, T., Gu, A., Pombra, J., Jabbari, S., Wu, S., & Lakkaraju, H. (2022). The disagreement problem in explainable machine learning: A practitioner's perspective. arXiv preprint arXiv:2202.01602.

Liu, L., Silva, E. A., Wu, C., & Wang, H. (2017). A machine learning-based method for the large-scale evaluation of the qualities of the urban environment. *Computers, Environment and Urban Systems*, *65*, 113–125. https://doi.org/10.1016/j.compenvurbsys.2017.06.003

Liu, Z., Wu, D., Liu, Y., Han, Z., Lun, L., Gao, J., Jin, G., & Cao, G. (2019). Accuracy analyses and model comparison of machine learning adopted in building energy consumption prediction. *Energy Exploration & Exploitation*, *37*(4), 1426–1451. https://doi.org/10.1177/0144598718822400

Lynch, K. (1960). *The image of the city*. MIT Press.

Mahajan, N., Patil, D., Kotkar, A., & Wasnik, K. (2019). Prediction of building structure age using machine learning. *International Journal of Advance Research*, *5*(3), 232–234.

Mishra, M. (2021). Machine learning techniques for structural health monitoring of heritage buildings: A state-of-the-art review and case studies. *Journal of Cultural Heritage*, 47, 227-245.

Mohammed, M., Khan, M. B., & Bashier, E. B. M. (2016). Machine learning: Algorithms and applications. CRC Press.

Myrtveit, I., Stensrud, E., & Shepperd, M. (2005). Reliability and validity in comparative studies of software prediction models. *IEEE Transactions on Software Engineering*, *31*(5), 380-391.

Narayanganj district. (2023, November 9). Bangladesh national information portal. Retrieved November 9, 2023, from https://www.dhaka.gov.bd/bn

Narkhede, S. (2018, May 9). *Understanding confusion matrix*. Medium. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Novack, T., Esch, T., Kux, H., & Stilla, U. (2011). Machine learning comparison between WorldView-2 and QuickBird-2-simulated imagery regarding object-based urban land cover classification. *Remote Sensing*, *3*(10), 2263–2282. https://doi.org/10.3390/rs3102263

Pham, A. D., Ngo, N. T., Truong, T. T. H., Huynh, N. T., & Truong, N. S. (2020). Predicting energy consumption in multiple buildings using machine learning for improving energy efficiency and sustainability. *Journal of Cleaner Production*, *260*, 121082.

Rahman, M. N., Rony, M. R. H., Jannat, F. A., Pal, S. C., Islam, M. S., Alam, E., & Islam, A. R. M. T. (2022). Impact of urbanization on urban heat island intensity in major districts of Bangladesh using remote sensing and geo-spatial tools. *Climate*, *10*(1), 3. https://doi.org/10.3390/cli10010003

Rajdhani Unnayan Kartipakkha. (2023, November 9). Retrieved November 9, 2023, from https://rajuk.gov.bd/site/page/68c8d4af-f493-43de-a54c-b0dc83d56bff/

RAJUK. (2016). *Detail Area Plan (DAP) 2016-2035*. RAJUK.

Ramírez, T., Hurtubia, R., Lobel, H., & Rossetti, T. (2021). Measuring heterogeneous perception of urban space with massive data and machine learning: An application to safety. *Landscape and Urban Planning*, *208*, 104002. https://doi.org/10.1016/j.landurbplan.2020.104002

Rizwan, M., & Anderson, D. V. (2018). Investigation on adaptive data condensation for exemplar based method in speech task. *2017 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2017 - Proceedings*, *2018-January*, 66–70. https://doi.org/10.1109/GLOBALSIP.2017.8308605

Robert, C. (2014). [Review of the book *Machine learning, a probabilistic perspective* by K. P. Murphy]. *CHANCE*, *27*(2), 62–63. https://doi.org/10.1080/09332480.2014.914768

Ruggieri, S. (2002). Efficient C4.5 [classification algorithm]. *IEEE Transactions on Knowledge and Data Engineering*, *14*(2), 438–444. https://doi.org/10.1109/69.991727

Russell, A. D., & Wong, W. C. M. (1993). New generation of planning structures. *Journal of Construction Engineering and Management*, *119*(2), 196–214. https://doi.org/10.1061/(ASCE)0733-9364(1993)119:2(196)

Saeed, R. M., & Abdulmohsin, H. A. (2023). A study on predicting crime rates through machine learning and data mining using text. *Journal of Intelligent Systems, 32*(1), 20220223. https://doi.org/10.1515/jisys-2022-0223

Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, *2*(3), 160. https://doi.org/10.1007/s42979-021-00592-x

Shafizadeh-Moghadam, H., Asghari, A., Tayyebi, A., & Taleai, M. (2017). Coupling machine learning, tree-based and statistical models with cellular automata to simulate urban growth. *Computers, Environment and Urban Systems*, *64*, 297–308. https://doi.org/10.1016/j.compenvurbsys.2017.04.002

Shen, J., Liu, C., Ren, Y., & Zheng, H. (2020). Machine learning assisted urban filling. In D. Holzer, W. Nakapan, A. Globa, I. Koh (Eds.), *RE: Anthropocene, Design in the Age of Humans - Proceedings of the 25th CAADRIA Conference*, Volume 2, (pp. 679-688). Association for Computer-Aided Architectural Design Research in Asia. https://doi.org/10.52842/conf.caadria.2020.2.679

Somvanshi, M., Chavan, P., Tambade, S., & Shinde, S.V. (2016). A review of machine learning techniques using decision tree and support vector machine. In *2016 International Conference on Computing Communication Control and automation (ICCUBEA)* (pp. 1-7). IEEE. doi: 10.1109/ICCUBEA.2016.7860040

Sun, H., Burton, H. V., & Huang, H. (2021). Machine learning applications for building structural design and performance assessment: State-of-the-art review. *Journal of Building Engineering*, *33*, 101816. https://doi.org/10.1016/j.jobe.2020.101816

World population by year (2023). Worldometer. https://www.worldometers.info/world-population/world-population-by-year/

Zhang, F., Zhou, B., Liu, L., Liu, Y., Fung, H. H., Lin, H., & Ratti, C. (2018). Measuring human perceptions of a large-scale urban region using machine learning. *Landscape and Urban Planning*, *180*, 148–160. https://doi.org/10.1016/j.landurbplan.2018.08.020

Zhang, Y., Burton, H. V., Sun, H., & Shokrabadi, M. (2018). A machine learning framework for assessing post-earthquake structural safety. *Structural Safety*, 72, 1-16. https://doi.org/10.1016/j.strusafe.2017.12.001

Appendix: Source code of model

This model was run with the Python codes below on Jupyter Notebook.

```
[1]: #Importing all the libraries
     import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     from sklearn.model_selection import train_test_split
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.preprocessing import LabelEncoder
     from sklearn.metrics import accuracy_score
     from sklearn.model_selection import learning_curve
     from functions import *

[2]: #Importing Data
     df = pd.read_csv("Structure_Use")
     df.head()

[3]: #Data cleaning
     df["Floor"].

[4]: df = df[df.Floor != ' ']

[5]: df["Floor"] = pd.to_numeric(df["Floor"])

[6]: df["Basement"].unique()

[7]: df = df[df.Basement != '21']

[8]: df["Basement"].replace(' ', "0", inplace=True)

[9]: df["Basement"] = pd.to_numeric(df["Basement"])

[10]: df["Structure_age"].unique()

[11]: df["Structure_age"].replace('15-20', '11-20', inplace=True)
      df["Structure_age"].replace('10-15', '11-20', inplace=True)
      df["Structure_age"].replace('20', '11-20', inplace=True)
      df["Structure_age"].replace('0-10m', '0-10', inplace=True)
      df["Structure_age"].replace('0-11', '0-10', inplace=True)

[12]: df = df[df.Structure_age != "50+"]

[13]: df = df[df.Structure_age != ' 50+']

[14]: df['Structure_age'] = df['Structure_age'].str.replace(' ', '', 1)

[15]: df["Dewlling_unit"].unique()

[16]: df = df[df.Dewlling_unit != ' ']

[17]: df["Dewlling_unit"] = pd.to_numeric(df["Dewlling_unit"])

[18]: df["Structure_type"].unique()

[19]: df["Structure_type"].replace('Under Con*', 'Under Construction', inplace=True)

[20]: df['Structure_type'] = df['Structure_type'].str.replace(' ', '', 1)

[21]: df["Structure_use"].unique()

[22]: df['Structure_use'] = df['Structure_use'].str.replace(' ', '', 1)

[23]: #Data Size
      df.shape
```

[24]:
```
#Spliting Data as dependent and independent variable
y = df["Structure_use"]
df_new = df.drop("Structure_use", axis="columns")
df_new.head()
```

[25]:
```
#Encoding all the values
le_Floor = LabelEncoder()
le_Basement = LabelEncoder()
le_Structure_age = LabelEncoder()
```

[26]:
```
df_new["Floor"] = le_Floor.fit_transform(df_new["Floor"])
df_new["Basement"] = le_Basement.fit_transform(df_new["Basement"])
df_new["Structure_age"] = le_Structure_age.
 ↪fit_transform(df_new["Structure_age"])
df_new["Dewlling_unit"] = le_Dewlling_unit.fit_transform(df["Dewlling_unit"])
df_new["Structure_type"] = le_Structure_type.fit_transform(df["Structure_type"])
```

[27]:
```
df_new.head()
```

[28]:
```
#Developing the Model
model = tree.DecisionTreeClassifier(criterion="entropy", random_state=42,␣
 ↪max_depth=20, min_samples_leaf=15)
```

[29]:
```
#Spliting Data for Training and Testing
X_train, X_test, y_train, y_test = train_test_split(df_new, y, test_size=0.2)
```

[30]:
```
#Fitting the model
model.fit(df_new, y)
```

[31]:
```
#Model accuray
print("Accuracy is for train sample", model.score(X_train, y_train)*100, "%")
print("Accuracy for test sample ", model.score(X_test, y_test)*100, "%")
```

[32]:
```
#This provide the insights of encoding
print("Values of Floor", pd.DataFrame(le_Floor.classes_))
print("Values of Basement", pd.DataFrame(le_Basement.classes_))
print("Values of Structure_age", pd.DataFrame(le_Structure_age.classes_))
print("Values of Dewlling_unit", pd.DataFrame(le_Dewlling_unit.classes_))
print("Values of Structure_type", pd.DataFrame(le_Structure_type.classes_))
```

[33]:
```
#Prediction
model.predict([[23, 3, 2, 122, 1]])
```

[34]:
```
# Generate learning curves
train_sizes, train_scores, test_scores = learning_curve(model, df_new, y, cv=5,␣
 ↪train_sizes=np.linspace(0.1, 1.0, 10))
```

[35]:
```
train_scores_mean = np.mean(train_scores, axis=1)
train_scores_std = np.std(train_scores, axis=1)
test_scores_mean = np.mean(test_scores, axis=1)
test_scores_std = np.std(test_scores, axis=1)
```

[36]:
```
# Plot the learning curves
plt.figure(figsize=(10, 6))
plt.plot(train_sizes, train_scores_mean,
plt.plot(train_sizes, test_scores_mean,
plt.fill_between(train_sizes, train_scores_mean
 →train_scores_mean + train_scores_std, alpha
plt.fill_between(train_sizes, test_scores_mean
 →test_scores_mean + test_scores_std, alpha
plt.xlabel("Training Set Size")
plt.ylabel("Accuracy Score")
plt.title("Learning Curves for Decision Tree")
plt.legend(loc="best")
plt.grid()
plt.show()
```