IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.17, No.1, January 2023, pp. 11~22
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.78537

Implementation of Ensemble Methods on Classification of CDK2 Inhibitor as Anti-Cancer Agent

Isman Kurniawan*[1], Mela Mai Anggraini[2], Annisa Aditsania[3], Erwin Budi Setiawan[4]
School of Computing, Telkom University, Bandung, Indonesia
e-mail: *[1]**ismankrn@telkomuniversity.ac.id**,
[2]melamaianggraini@students.telkomuniversity.ac.id, [3]aaditsania@telkomuniversity.ac.id,
[4]erwinbudisetiawan@telkomuniversity.ac.id

Abstrak

Kanker dikenal sebagai penyebab kematian nomor dua di dunia. Sekitar 7-10 juta kasus kematian akibat kanker terjadi setiap tahun. Pengobatan terbaru untuk menyembuhkan kanker adalah kemoterapi. Namun, pengobatan kemoterapi diketahui memiliki efek samping dan masalah resistensi sel terhadap obat-obatan tertentu. Oleh karena itu, diperlukan pengembangan obat baru yang dapat mengurangi efek samping dan memberikan efek pengobatan yang lebih baik. Secara umum, obat anti kanker dikembangkan dengan menargetkan enzim Cyclin-Dependent Kinase 2 (CDK2). Desain obat konvensional tidak efektif dan efisien untuk mendapatkan calon obat baru karena tidak adanya informasi tentang aktivitas biologis sebelum disintesis. Dalam penelitian ini, kami bertujuan untuk mengembangkan model untuk memprediksi aktivitas inhibitor CDK2 dengan menggunakan metode ensemble, yaitu XGBoost, Random Forest, dan AdaBoost. Penelitian dilakukan dengan menghitung beberapa fitur fingerprint yaitu Estate, Extended, Maccs, dan Pubchem sebagai variabel fitur. Berdasarkan hasil tersebut, kami menemukan bahwa Random Forest dengan fingerprint Pubchem memberikan hasil terbaik dengan nilai Matthews Correlation Coefficient (MCC) dan Area Under the ROC Curve (AUC) masing-masing adalah 0.979 dan 0.999. Dari studi ini, kami telah berkontribusi untuk menunjukkan potensi metode ensemble dengan fitur fingerprint untuk prediksi bioaktivitas, khususnya inhibitor CDK2 sebagai agen anti kanker.

Kata kunci*—QSAR, CDK2, XGBoost, random forest, AdaBoost*

Abstract

Cancer is known as the second leading cause of death worldwide. About 7-10 million cases of death by cancer occur every year. The recent treatment to heal the cancer is chemotherapy. However, chemotherapy treatment is known to have side effects and cell resistance issues to certain drugs. Therefore, it is required to develop a new drug that can reduce the side effects and provide a better treatment effect. In general, anti-cancer drugs are developed by targeting Cyclin-Dependent Kinase 2 (CDK2) enzyme. Conventional drug design is not effective and efficient for obtaining new drug candidates because of no information about the biological activity before it is synthesized. In this study, we aim to develop a model to predict the activity of CDK2 inhibitors by using ensemble methods, i.e., XGBoost, Random Forest, and AdaBoost. The study was conducted by calculating several fingerprints, i.e., Estate, Extended, Maccs, and Pubchem, as feature variables. Based on the results, we found that Random Forest with Pubchem fingerprint gives the best result with the value of Matthews Correlation Coefficient (MCC) and Area Under the ROC Curve (AUC) values are 0.979 and 0.999, respectively. From this study, we contributed to revealing the potency of the ensemble with fingerprint in bioactivity prediction, especially CDK2 inhibitors as anti-cancer agents.

Keywords—*QSAR, CDK2, XGBoost, random forest, AdaBoost*

Received October 21th,2022; Revised November 11th, 2022; Accepted January 30th, 2023

1. INTRODUCTION

Cancer is the second leading cause of death worldwide [1]. As time and population increase, the number of people with cancer continues to increase [2]. Every year, about 7-10 million cases of cancer deaths occur worldwide [2]. Cancer is a disease caused by the growth of abnormal cells that cause damage to the tissues of the human body. Cancer can affect people of all ages, both men and women. The most common types of cancer suffered by women include cervical, breast, ovarian, skin, thyroid, colorectal, lymph node, uterine, colon, and nasopharyngeal cancers [3]. The most common types of cancer suffered by men include prostate cancer, lung cancer, and liver cancer [4].

Currently, one of the treatments commonly used to treat cancer is chemotherapy. In the process, chemotherapy not only damages cancer cells but also damages other healthy cells [5]. Therefore, chemotherapy treatment in the long term can be harmful to the body. In addition, chemotherapy also has side effects that can affect physical health, life quality, and also emotions [6]–[8]. The most common side effect of chemotherapy treatment is fatigue, followed by diarrhea and constipation [9]. The effectiveness of chemotherapy itself is limited by the resistance of cells to certain types of drugs. This resistance can be caused by mutations that arise during chemotherapy treatment or through various other adaptive responses, such as increased expression of therapeutic targets and activation of alternative compensatory signaling pathways [10].

Regarding the resistance problem, it is required to develop a new drug that can reduce side effects and provide better treatment effects. In general, anti-cancer drugs are developed by considering Cyclin-Dependent Kinase 2 (CDK2) as the target. CDK2 enzymes are part of Cyclin-Dependent Kinases (CDKs), which play an important role in the growth of cancer cells. This relates to the role of this enzyme in the regulation of complex processes of the cell division cycle, apoptosis, transcription, and differentiation [11], [12]. Conventional drug designs are known to be ineffective and inefficient because it is necessary to synthesize the compound to know the activity [13]. Therefore, the drug design can be accelerated by implementing a machine learning model to predict the activity. The prediction process can be done using a mathematical model to determine the correlation between structure and activity, known as Quantitative Structure-Activity Relationships (QSAR).

In 2006, Singh and coworkers performed a 3D-QSAR CoMFA study of CDK2 and CDK4 inhibitors [14]. They found strong correlative and predictive abilities with conventional and predictive correlation coefficients against the CDK4 target are 0.913 and 0.760, respectively, and those values against the CDK2 target are 0.941 and 0.765, respectively [14]. In 2006, Singh and coworkers also conducted a 3D-QSAR CoMFA study of CDK1 and CDK2 inhibitors on oxindole compounds as inhibitors [15]. The results show that the compound has better correlative and predictive abilities against CDK2 than CDK1 [15]. In 2010, Lan and coworkers conducted a 3D-QSAR study using the CoMFA method and docking on a series of pyrazolo[4,3-h]quinazoline-3-carboxamides compounds as CDK2 inhibitors [16]. The results provide a useful guideline for the rational design of new CDK2 inhibitors [16].

This study aims to implement the ensemble method for classifying CDK2 inhibitors as anti-cancer agents. Also, the effect of fingerprint techniques as a feature variable on the performance of the model prediction is investigated. The ensemble methods used in this study are XGBoost, Random Forest, and AdaBoost. XGBoost is known to be able to support various objective functions such as regression, classification, and others [17]. One of the advantages of XGBoost is that it can have many parameters that can be adjusted to make good predictions [18]. In addition, the XGBoost system runs ten times faster than other methods [17]. Random forest is an ensemble learning model that uses bagging as a learning method [19]. The random forest can handle missing values and varied feature types and is suitable for modeling high-dimensional data. Meanwhile, Adaptive Boosting (AdaBoost) is an algorithm that uses boosting as its learning. Boosting is a learning technique by combining weak learners by adjusting the weights through a repetition process. This study helped to expose the strength of the ensemble with fingerprint in bioactivity prediction, particularly for CDK2 inhibitors as anti-cancer drugs.

2. METHODS

2.1 Data Set

The data set used in this study consists of the chemical structure and IC50 value of CDK2 inhibitors retrieved from ChemBL, which contained 2,328 compounds [20]. IC50 is a biological activity that represents the amount of drug required to inhibit half of the target activity. The data set was divided into two sub-data using IC50 as the classification criteria. The first sub-data contains active inhibitors with IC50 values less than equal to 10 µM, and the second sub-data contains inactive inhibitors with IC50 values greater than 500 µM. Meanwhile, data with IC50 values between 10µM to 500µM are omitted. Active inhibitors were labeled with a value of 1, while inactive inhibitors were labeled with a value of 0. The number of active and inactive inhibitors is 1164 and 36, respectively. Due to the unbalanced number of classes, we collected putative compounds to balance the amount of data. This process was done by making several clusters from a large collection of compounds. Then, the putative compound was selected from compounds from a cluster that is not contained active inhibitors. After adding the putative compound to the inactive class, we obtain a similar number of data involved in active and inactive classes.

Furthermore, the molecular descriptor as the feature variable was calculated from the compound structure. The 3-dimensional (3D) structure of the compound was obtained by converting the Simplified Molecular-Input Line-Entry System (SMILES) notation to Structure-Data File (SDF) using Open Babel [21]. We calculated the fingerprint representation of a compound as the molecular descriptor, in which the fingerprint is represented in 4 forms, i.e., Estate, Extended, MACCS, and Pubchem. Each fingerprint represents fragments contained in the structure differently and produces a different bit number of the fingerprint. The total number of bits contained in Estate, Extended, MACCS, and Pubchem fingerprints are 80, 1025, 167, and 882, respectively. Furthermore, the data is randomly divided into two sub-data, namely training data and test data, with a ratio of 4:1 [22]. The number of samples in the train and test sets is 1862 and 466, respectively, while the number of active and inactive classes in both sets is provided in Figure 1.

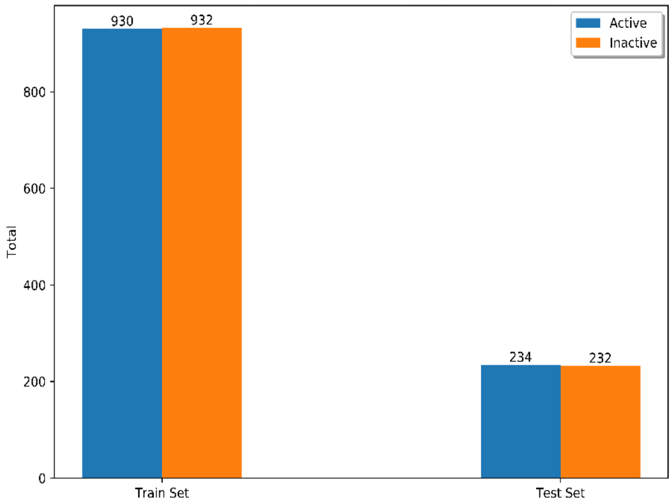


Figure 1 The number of active and inactive classes in the train and test set

2.2 Methods

2. 2.1 XGBoost

Extreme Gradient Boosting (XGBoost) is an improved algorithm based on gradient-boosting decision trees that can build boosted trees efficiently and operate in parallel [23]. One of the advantages of XGBoost is that the algorithm has many parameters that can be adjusted to make good predictions [18]. XGBoost can be implemented to solve regression and classification problems based on the Gradient Boosting Decision Tree (GBDT) [17]. The loss function equation is the difference between regression and classification in XGBoost in tree construction. In the regression case, the loss function formula used can be seen in Equation 1.

$$\sum_{i=1}^{n} L(y_i, p_i) = \frac{1}{2}(y_i - p_i) \tag{1}$$

where n is the number of observed data, y_i is the value of the observed i-th data, and p_i is the predictive probability value of the i-th observation data. Meanwhile, in the case of the loss function equation classification, it can be seen in Equation 2.

$$L(y_i, p_i) = -[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{2}$$

where y_i is the value of the observed i-th data and p_i is the i-th data prediction probability value. From Equation 2, by performing the derivative (d) of the formula, we can get the equation Gradient (g_i) in the first derivation and Hessian (h_i) in the second derivation. The Gradient and Hessian equations obtained can be seen in Equations 3 and 4.

$$g_i = \frac{d}{d\log(odds)} L(y_i, \log(odds)_i) = -(y_i - p_i) \tag{3}$$

$$\begin{aligned} h_i &= \frac{d^2}{d\ \log(odds)^2} L(y_i, \log(odds)_i) \\ &= p_i \times (1 - p_i) \end{aligned} \tag{4}$$

where

$$odds = \left(\frac{p}{1-p}\right) \tag{5}$$

We can substitute Equations 3 and 4 into the Output Value (O_value) equation to calculate the optimal output value.

$$O_{value} = -\frac{(g_1+g_2+g_3+\cdots+g_n)}{(h_1+h_2+h_3+\cdots+h_n+\lambda)} \tag{6}$$

$$O_{value} = \frac{(\sum Residual_i)}{\sum[Previous\ Probability_i \times (1 - Previous\ Probability_i)] - \lambda} \tag{7}$$

Equation 6 is O_{value} equation before being converted to Equations 3 and 4. After conversion, the final equation for O_{value} is obtained as in Equation 7. Furthermore, to develop the tree that is built, it is necessary to calculate the similarity value. The similarity equation can be seen in Equation 8.

$$Similarity = \frac{(\sum Residual_i)^2}{\sum[Previous\ Probability_i \times (1 - Previous\ Probability_i)] - \lambda} \tag{8}$$

Then, the Gain value is calculated to determine which leaves/branches will be used on the tree. Leaves/Branches to be used are those with the greatest Gain value. The parameters of XGBoost used in this study are presented in Table 1.

Table 1 The Parameters of XGBoost

Parameters	Values
gamma	0
learning_rate	0.1
max_delta_step	0
max_depth	3
min_child_weight	1
n_estimators	100
random_state	0

2. 2.2 Random Forest

Random forest (RF) is an ensemble learning that uses bagging as a learning method [19]. RF increase tree diversity by having them grow from different subsets of training data created through bagging or bootstrap aggregating. This algorithm can handle missing values and various variables (continuous, binary, categorical) and is suitable for modeling high-dimensional data. With the ensemble scheme and bootstrapping, RF can overcome the problem of overfitting. RF can measure the importance of each feature by training the model. In the process, the random forest directly selects the features while the classification rules are made. A commonly used measure of importance from a random forest is the Gini Importance. Gini is a popular choice used in biological data mining tasks [24]. Gini Importance is directly derived from the Gini Index [25]. The way the random forest works is, first, making a random sample from the training data. Then, the random forest will make a decision tree for each sample and get the prediction results from each decision tree made. In the resulting random forest decision tree, the random forest classifier uses a split function called the Gini Index to determine which attributes should be split during the tree learning phase. The Gini Index measures the degree of impurity or inequality of the sample assigned to a node based on the split in its parent. The Gini Index equation is shown in Equation 9.

$$Gini = \sum_{I=1}^{C} f_i(1 - f_i) \tag{9}$$

where f_i is the frequency of the i-th label on a node and C is the number of unique labels.

Table 2 The Parameters of Random Forest

Parameters	Values
criterion	‘gini’
max_depth	None
max_features	Auto’
max_leaf_nodes	None
min_impurity_decrease	0
min_impurity_split	None
min_samples_leaf	1
min_samples_split	2
min_weight_fraction_leaf	0
n_estimators	10
verbose	0
warm_start	FALSE

2. 2.3 AdaBoost

Adaptive Boosting (AdaBoost) is an algorithm that uses boosting approach in the learning process. Boosting is a learning technique that combines several weak learners to produce more accurate predictions. The AdaBoost algorithm discovered by Freund and Schapire combines weak learners by adjusting the weights through repetition and makes AdaBoost able to produce accurate predictive models [26]. The AdaBoost algorithm works as follows. First, the sample weights were initialized for each training data using Equation 10.

$$Sample\ weight = \frac{1}{n} \tag{10}$$

where n is the total training data. Secondly, a decision tree was built with each feature, classifying the data, and evaluating the results. Thirdly, the predicted result label is compared with the actual label. The tree with the best performance in classifying the sample will be used in the next iteration. Fourthly, the significance of the tree's performance in the final classification will be calculated by using Equation 11.

$$Significance = \frac{1}{2}\log\left(\frac{1-Total_{error}}{Total_{error}}\right) \tag{11}$$

where $Total_{error}$ is the number of sample weights that are classified as incorrect. Fifthly, update the sample weights so that the next decision tree will consider the errors made by the previous decision tree. For trees classified incorrectly, Previously updated weights for trees that are classified incorrectly and correctly were calculated using Equations 12 and 13, respectively.

$$New\ sample\ weight = sample\ weight\ \times\ e^{significance} \tag{12}$$

$$New\ sample\ weight = sample\ weight\ \times\ e^{-significance} \tag{13}$$

Sixthly, a new data set was created with a similar size to the original data by picking the data randomly. The second to the sixth step was repeated until the maximum iteration was reached. Finally, a set of decision trees that have been created was used to make predictions on the test data by dividing the tree into two groups based on the results of each tree's decisions. Then, the amount of significance for each tree in the group was calculated. The final classification is determined according to the largest number of significance. The parameters of AdaBoost used in this study are presented in Table 3.

Table 3 The Parameters of AdaBoost

Parameters	**Values**
algorithm	‘Samme.R’
base_estimator	None
learning_rate	1
n_estimators	50
random_state	None

2. 2.4 Decision Tree

The Decision Tree (DT) is a classification method that recursively breaks down a data set into smaller sub-data [27]. DT can handle non-linear relationships between features and classes [27]. In addition, DT can work flexibly and efficiently in computing [27]. Each DT consists of a root node and a leaf node. Each leaf node has only one root node, and the leaf node refers to the class label. In the process, DT works sequentially in testing the data. The most common algorithms used in DT are ID3 and classification and regression trees (CART). ID3 is a very simple decision tree algorithm [28]. ID3 uses the information gain separation function as the separation criterion. The information gain equation can be seen in Equation 14.

$$Information\ Gain\ (X,a) = E(X) - E(X \mid a) \tag{14}$$

where $Information\ Gain\ (X,a)$ is the value of information gain of dataset X for variable a.

$E(X)$ is the entropy value for the dataset X before the change and $E(X \mid a)$ is the conditional entropy for the dataset given the variable a. The Entropy equation can be seen in Equation 15.

$$E = \sum_{i}^{C} p_i log_2 p_i \tag{15}$$

where p_i is the probability of choosing an element of class i at random and C is the class in the data. The tree expansion will stop when all the instances have a residual value of the target feature or when the information gain is not more than 0. In addition, ID3 does not apply a pruning procedure and cannot handle numeric attributes or missing values. CART is a decision tree algorithm that handles binary cases, which divides a single variable at each node [29]. CART uses the Gini Index separation function as the separation criterion. The Gini Index equation can be seen in Equation 9.

2. 3 Model Validation

We validate the model by evaluating several validation parameters derived from the confusion matrix, as shown in Table 4. The validation parameters consist of Sensitivity (SE), Specificity (SP), Precision (PREC), Accuracy (Q), F1-Score, and Matthews Correlation Coefficient (MCC). We used MCC as an overall measurement to determine the model that gives the best performance. MCC is a coefficient that represents the correlation between the observed and predicted binary classifications. MCC will return values between -1 and 1, where the correlation coefficient value of 1 represents a correct prediction and a coefficient value of -1 is a false prediction [30]. Those validation parameters are evaluated by using Equation (16) - (21).

Table 4 Confusion Matrix

Class	Class 1 (Predictive)	Class 2 (Actual)
Class 1 (Actual)	TP (True Positive)	FN (False Negative)
Class 2 (Predictive)	FP (False Positive)	TN (True Negative)

$$SE = \frac{TP}{TP+FN} \tag{16}$$

$$SP = \frac{TN}{TN+FP} \tag{17}$$

$$PREC = \frac{TP}{TP+FP} \tag{18}$$

$$Q = \frac{TP+TN}{TP+FN+TN+FP} \tag{19}$$

$$\text{F1} - Score = 2 \times \frac{PREC \times SE}{PREC + SE} \tag{20}$$

$$MCC = \frac{(TP \times TN)-(FP \times FN)}{\sqrt{(TP+FN)}(TP+FP)(TN+FN)(TN+FP)} \tag{21}$$

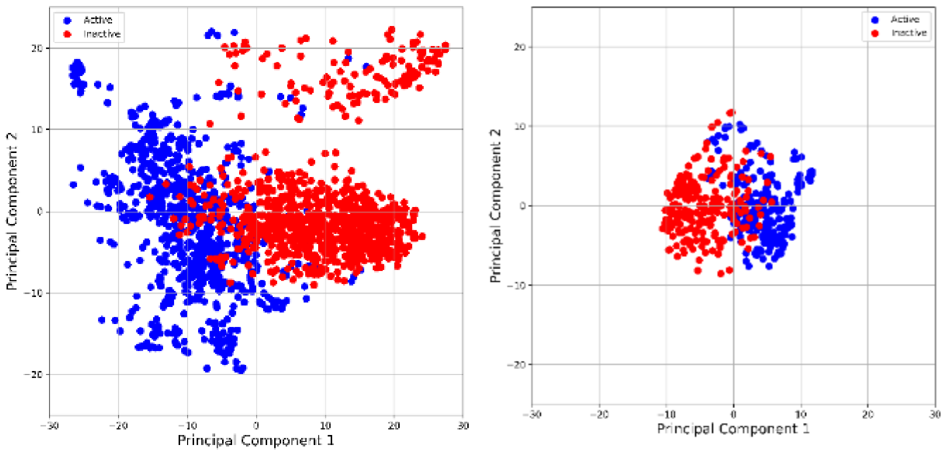
In addition, the model is also evaluated with the Receiver Operating Characteristics (ROC) curve. This curve describes the success rate of predictions and errors in the classification model. ROC is plotted by taking True Positive and False Negative values on the x-axis and y-axis. From this curve, we can also calculate the Area Under the ROC Curve (AUC) parameter [31]. The AUC measures the model's ability to differentiate between the two classification groups and thus represents predictive accuracy.

After getting the best model, a Y-Scrambling experiment was carried out on the model to prove that the model did not match the coincidence correlation. Ten random models were developed by randomizing the target value while maintaining the descriptor. The performance of the random model is evaluated by calculating the MCC value and comparing the value with the non-random model.

3. RESULTS AND DISCUSSIONS

3. 1 Data Distribution

We investigated the distribution To investigate the distribution of active and inactive inhibitors in the train and test set, we derived two Principal Components from the set of features using Principal Component Analysis (PCA). PCA is a data simplification technique that transforms the data linearly to form a new coordinate system with maximum variance. The distribution of the data obtained from PCA analysis can be seen in Figure 2. We found that the distribution of active and inactive classes in the train and test set is distinguishable that indicating that the data set is not too complex. However, an overlap of data of both classes is also found in the center of the graph which points out the challenge to classify those samples.



(a) (b)

Figure 2 The distribution of active and inactive classes in (a) train and (b) test set

3. 2 Model Validation

We developed model prediction by utilizing three ensemble methods, i.e., XGBoost, Random Forest, and AdaBoost, with Decision Tree as the baseline method, and four fingerprint feature types, i.e., Estate, Extended, MACCS, and Pubchem. Those models were developed by using the parameters provided in Tables 1, 2, and 3. Then, the performance of those models was evaluated by calculating several validation parameters on the train and test set, as shown in Tables 5 and 6, respectively.

As for the train set, we found that the performance of all models is quite satisfying, which is indicated by the value of the validation parameters. However, the Decision Tree (DT) shows the best performance on all fingerprints with a perfect value of all validation parameters in three fingerprint types, i.e., fp_extended, fp_maccs, and fp_pubchem. This indicates the ability of the DT to predict the train set perfectly. However, the good performance of a model on the train set did not guarantee that the model could give a similar performance on the test set. Furthermore, the results might lead to an overfitting condition.

Table 5 The Validation Results of the Train Set

Fingerprint	Method	Q	SE	SP	PREC	F1-Score	MCC	AUC
fp_estate	XGBoost	0.952	0.958	0.947	0.946	0.952	0.904	0.990
	Random Forest	0.996	0.997	0.995	0.995	0.996	0.991	**1.000**
	AdaBoost	0.931	0.946	0.918	0.915	0.930	0.863	0.978
	DT	0.998	0.999	0.997	0.997	0.998	0.996	**1.000**
fp_extended	XGBoost	0.999	0.999	**1.000**	**1.000**	0.999	0.999	**1.000**
	Random Forest	0.999	0.998	1.000	1.000	0.999	0.998	1.000
	AdaBoost	0.990	0.989	0.991	0.991	0.990	0.981	1.000
	DT	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
fp_maccs	XGBoost	0.988	0.995	0.981	0.981	0.988	0.975	0.999
	Random Forest	0.999	**1.000**	0.999	0.999	0.999	0.999	**1.000**
	AdaBoost	0.971	0.978	0.964	0.963	0.971	0.942	0.997
	DT	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
fp_pubchem	XGBoost	0.988	0.995	0.981	0.981	0.988	0.975	0.999
	Random Forest	0.996	0.994	0.999	0.999	0.996	0.992	**1.000**
	AdaBoost	0.983	0.984	0.982	0.982	0.983	0.966	0.999
	DT	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**

As for the test set, we found that the combination of Random Forest and Pubchem fingerprint gives the best results on all validation parameters, except selectivity (SE). This indicates the suitability of the utilization of Pubchem fingerprint on the bagging scheme of Random Forest. Meanwhile, the combination of XGBoost and Pubchem fingerprint also gives the best result on selectivity and AUC. According to the value of MCC, we found that the best result is obtained from the RF-Pubchem method with an MCC value is 0.979. Meanwhile, we found that the combination of AdaBoost and Estate fingerprint gives the worst result with the value of MCC being 0.817. This indicates that the boosting scheme of AdaBoost is not suitable for the feature generated by the Estate fingerprint.

Table 6 The Validation Results of the Test Set

Fingerprint	Method	Q	SE	SP	PREC	F1-Score	MCC	AUC
fp_estate	XGBoost	0.938	0.948	0.928	0.927	0.937	0.876	0.987
	Random Forest	0.948	0.941	0.956	0.957	0.949	0.897	0.992
	AdaBoost	0.908	0.932	0.886	0.880	0.905	0.817	0.961
	DT	0.942	0.941	0.943	0.937	0.939	0.884	0.941
fp_extended	XGBoost	0.981	0.991	0.970	0.970	0.981	0.962	0.999
	Random Forest	0.979	0.987	0.970	0.970	0.978	0.957	0.996
	AdaBoost	0.968	0.987	0.950	0.949	0.967	0.936	0.993
	DT	0.957	0.959	0.955	0.951	0.955	0.914	0.957
fp_maccs	XGBoost	0.970	0.978	0.962	0.962	0.970	0.940	0.996
	Random Forest	0.970	0.970	0.970	0.970	0.970	0.940	0.993
	AdaBoost	0.953	0.973	0.934	0.932	0.952	0.906	0.984
	DT	0.972	0.977	0.967	0.964	0.971	0.944	0.972
fp_pubchem	XGBoost	0.985	**0.996**	0.975	0.974	0.985	0.970	**0.999**
	Random Forest	**0.989**	0.991	**0.987**	**0.987**	**0.989**	**0.979**	**0.999**
	AdaBoost	0.970	0.995	0.947	0.944	0.969	0.941	0.991
	DT	0.957	0.955	0.959	0.955	0.955	0.914	0.957

We provided the average value of the validation parameters for each method to compare the performance, as shown in Table 7. As for the train set, we found that the Decision Tree (DT) method give the best results for all validation parameter. Meanwhile, AdaBoost gives the worst results indicated by the lowest value of MCC. However, as for the test set, we found that Random Forest gives the best performance with the highest value on all validation parameters. On the contrary, the worst results were obtained from the Decision Tree with the lowest value on almost all validation parameters. These results point out the overfitting situation of the Decision Tree method that might be caused by the too-complex tree structure of the method.

Regarding the contribution of fingerprint type, we presented the average value of the validation parameter for each fingerprint type, as shown in Table 8. As for the train set, we found that Extended FP gives the best value for all validation parameters, which indicated the ability of this fingerprint to provide a good feature for the train set. Meanwhile, as for the test set, we found that the best results were obtained from the model Pubchem FP with the highest value for all validation parameters. This indicated that Pubchem FP gives a balanced quality of features for both the train and test set. We also provided the receiver operating characteristic (ROC) curve that was used to calculate the AUC parameter.

Table 7 The Average Values of the Validation Parameter for Each Method

Train Set							
Method	**Q**	**SE**	**SP**	**PREC**	**F1-Score**	**MCC**	**AUC**
XGBoost	0.982	0.986	0.977	0.977	0.982	0.964	0.997
Random Forest	0.995	0.997	0.998	0.998	0.998	0.995	1.000
AdaBoost	0.969	0.974	0.964	0.963	0.968	0.938	0.993
DT	**0.999**	**1.000**	**0.999**	**0.999**	**0.999**	**0.999**	**1.000**
Test Set							
Method	**Q**	**SE**	**SP**	**PREC**	**F1-Score**	**MCC**	**AUC**
XGBoost	0.968	0.978	0.959	0.958	0.968	0.937	0.995
Random Forest	**0.972**	**0.972**	**0.971**	**0.971**	**0.972**	**0.943**	**0.995**
AdaBoost	0.950	0.972	0.929	0.926	0.949	0.900	0.982
DT	0.957	0.958	0.956	0.952	0.955	0.914	0.957

Table 8 The Average Values of the Validation Parameter for Fingerprint Type

Train Set							
Method	**Q**	**SE**	**SP**	**PREC**	**F1-Score**	**MCC**	**AUC**
fp_estate	0.969	0.975	0.964	0.963	0.969	0.939	0.992
fp_extended	**0.997**	**0.997**	**0.998**	**0.998**	**0.997**	**0.994**	**1.000**
fp_maccs	0.990	0.993	0.986	0.986	0.989	0.979	0.999
fp_pubchem	0.992	0.993	0.990	0.990	0.992	0.983	1.000
Test Set							
Method	**Q**	**SE**	**SP**	**PREC**	**F1-Score**	**MCC**	**AUC**
fp_estate	0.931	0.940	0.923	0.922	0.931	0.863	0.980
fp_extended	0.967	0.977	0.958	0.957	0.966	0.935	0.982
fp_maccs	0.962	0.970	0.955	0.953	0.962	0.925	0.982
fp_pubchem	**0.979**	**0.990**	**0.969**	**0.968**	**0.979**	**0.959**	**0.990**

Finally, we evaluated the best model by using the y-scrambling method to make sure that the result is not related to coincidental correlation. The plot of the y-scrambling analysis is presented in Figure 3. We compared the performance of the model developed with the original data (no-random) with the model developed using 10 trials of shuffle data. We found that the MCC score of the original model outperformed compared to the shuffle one. This indicates that there is no coincidental correlation found in our model.

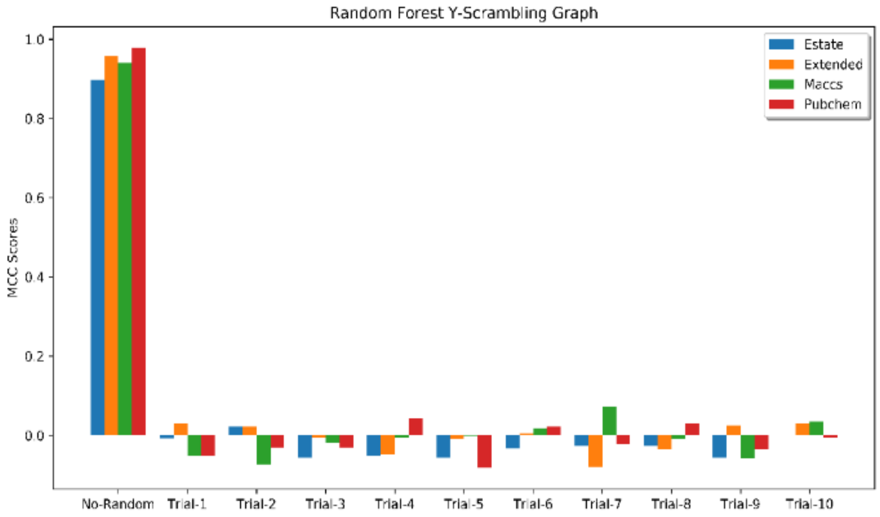


Figure 3 The results of the y-scrambling analysis

4. CONCLUSION

We have developed a prediction model of CDK2 inhibitor as an anti-cancer agent by using 3 ensemble methods, i.e XGBoost, Random Forest, dan AdaBoost, and 4 types of fingerprint features, i.e. Estate, Extended, MACCS, and Pubchem fingerprint. Based on the results, we found that the best model obtained from Random Forest with Pubchem fingerprint with the value of accuracy, F-1 score, MCC, and AUC are 0.989, 0.989, 0.979, and 0.999, respectively. To improve the results, we suggest combining those methods to become the weight-based majority voting method. Also, feature selection should be considered to be conducted to avoid too complex a model and overfitting conditions.

REFERENCES

[1] D. Lu, T.-R. Lu, and H. wu, “Personalized Cancer Therapy: A Perspective,” *Clin. Exp. Pharmacol.*, vol. 04, p. 153, Jan. 2014, doi: 10.4172/2161-1459.1000153.

[2] D. Lu, T.-R. Lu, J.-Y. Che, and N. sastry Yarla, “Individualized Cancer Therapy, What is the Next Generation?,” vol. 2, Jun. 2018.

[3] M. F. Aziz, “Gynecological cancer in Indonesia,” *J. Gynecol. Oncol.*, vol. 20, no. 1, pp. 8–10, Mar. 2009, doi: 10.3802/jgo.2009.20.1.8.

[4] “Cancer today.” [Online]. Available: http://gco.iarc.fr/today/home. [Accessed: Oct. 19, 2022]

[5] B. A. Chabner and T. G. Roberts, “Chemotherapy and the war on cancer,” *Nat. Rev. Cancer*, vol. 5, no. 1, pp. 65–72, Jan. 2005, doi: 10.1038/nrc1529.

[6] N. Carelle, E. Piotto, A. Bellanger, J. Germanaud, A. Thuillier, and D. Khayat, “Changing patient perceptions of the side effects of cancer chemotherapy,” *Cancer*, vol. 95, no. 1, pp. 155–163, Jul. 2002, doi: 10.1002/cncr.10630.

[7] A. Coates *et al.*, “On the receiving end--patient perception of the side-effects of cancer chemotherapy,” *Eur. J. Cancer Clin. Oncol.*, vol. 19, no. 2, pp. 203–208, Feb. 1983, doi: 10.1016/0277-5379(83)90418-2.

[8] M. de Boer-Dennert *et al.*, “Patient perceptions of the side-effects of chemotherapy: the influence of 5HT3 antagonists.,” *Br. J. Cancer*, vol. 76, no. 8, pp. 1055–1061, 1997.

[9] “American Cancer Society | Information and Resources about for Cancer: Breast, Colon, Lung, Prostate, Skin.” [Online]. Available: https://www.cancer.org. [Accessed: Oct. 19, 2022]

[10] D. B. Longley and P. G. Johnston, “Molecular mechanisms of drug resistance,” *J. Pathol.*, vol. 205, no. 2, pp. 275–292, Jan. 2005, doi: 10.1002/path.1706.

[11] K. Lingfei, Y. Pingzhang, L. Zhengguo, G. Jianhua, and Z. Yaowu, “A study on p16, pRb, cdk4 and cyclinD1 expression in non-small cell lung cancers,” *Cancer Lett.*, vol. 130, no. 1, pp. 93–101, Aug. 1998, doi: 10.1016/S0304-3835(98)00115-3.

[12] R. N. Rao, “Targets for cancer therapy in the cell cycle pathway,” *Curr. Opin. Oncol.*, vol. 8, no. 6, pp. 516–524, Nov. 1996.

[13] S. Vilar, G. Cozza, and S. Moro, “Medicinal chemistry and the molecular operating environment (MOE): application of QSAR and molecular docking to drug discovery,” *Curr. Top. Med. Chem.*, vol. 8, no. 18, pp. 1555–1572, 2008, doi: 10.2174/156802608786786624.

[14] S. K. Singh, N. Dessalew, and P. V. Bharatam, “3D-QSAR CoMFA study on indenopyrazole derivatives as cyclin dependent kinase 4 (CDK4) and cyclin dependent kinase 2 (CDK2) inhibitors,” *Eur. J. Med. Chem.*, vol. 41, no. 11, pp. 1310–1319, Nov. 2006, doi: 10.1016/j.ejmech.2006.06.010.

[15] S. K. Singh, N. Dessalew, and P. V. Bharatam, “3D-QSAR CoMFA study on oxindole derivatives as cyclin dependent kinase 1 (CDK1) and cyclin dependent kinase 2 (CDK2)

inhibitors," *Med. Chem. Shariqah United Arab Emir.*, vol. 3, no. 1, pp. 75–84, Jan. 2007, doi: 10.2174/157340607779317517.

[16] P. Lan, W.-N. Chen, G.-K. Xiao, P.-H. Sun, and W.-M. Chen, "3D-QSAR and docking studies on pyrazolo[4,3-h]qinazoline-3-carboxamides as cyclin-dependent kinase 2 (CDK2) inhibitors," *Bioorg. Med. Chem. Lett.*, vol. 20, no. 22, pp. 6764–6772, Nov. 2010, doi: 10.1016/j.bmcl.2010.08.131.

[17] T. Chen *et al.*, "xgboost: Extreme Gradient Boosting." Apr. 16, 2022 [Online]. Available: https://CRAN.R-project.org/package=xgboost. [Accessed: Oct. 19, 2022]

[18] R. P. Sheridan, W. M. Wang, A. Liaw, J. Ma, and E. M. Gifford, "Extreme Gradient Boosting as a Method for Quantitative Structure–Activity Relationships," *J. Chem. Inf. Model.*, vol. 56, no. 12, pp. 2353–2360, Dec. 2016, doi: 10.1021/acs.jcim.6b00591.

[19] Y. Qi, "Ensemble Machine Learning," pp. 307–323.

[20] "ChEMBL Database." [Online]. Available: https://www.ebi.ac.uk/chembl/. [Accessed: Jan. 08, 2020]

[21] N. M. O'Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, "Open Babel: An open chemical toolbox," *J. Cheminformatics*, vol. 3, no. 1, p. 33, Dec. 2011, doi: 10.1186/1758-2946-3-33.

[22] I. Kurniawan, M. Rosalinda, and N. Ikhsan, "Implementation of ensemble methods on QSAR Study of NS3 inhibitor activity as anti-dengue agent," *SAR QSAR Environ. Res.*, vol. 31, no. 6, pp. 477–492, Jun. 2020, doi: 10.1080/1062936X.2020.1773534.

[23] J. H. Friedman, "Stochastic gradient boosting," *Comput. Stat. Data Anal.*, vol. 38, no. 4, pp. 367–378, Feb. 2002, doi: 10.1016/S0167-9473(01)00065-2.

[24] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, Oct. 2007, doi: 10.1093/bioinformatics/btm344.

[25] Y. L. Pavlov, *Random Forests*. 2019.

[26] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Computational Learning Theory - 2nd European Conference, EuroCOLT 1995, Proceedings*, Jan. 1995, pp. 23–37, doi: 10.1007/3-540-59119-2_166 [Online]. Available: https://collaborate.princeton.edu/en/publications/a-decision-theoretic-generalization-of-on-line-learning-and-an-ap-2. [Accessed: Oct. 19, 2022]

[27] M. A. Friedl and C. E. Brodley, "Decision tree classification of land cover from remotely sensed data," *Remote Sens. Environ.*, vol. 61, no. 3, pp. 399–409, Sep. 1997, doi: 10.1016/S0034-4257(97)00049-7.

[28] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, Mar. 1986, doi: 10.1007/BF00116251.

[29] "Classification and Regression Trees | Leo Breiman | Taylor & Francis e." [Online]. Available: https://www.taylorfrancis.com/books/mono/10.1201/9781315139470/classification-regression-trees-leo-breiman. [Accessed: Oct. 19, 2022]

[30] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric," *PLOS ONE*, vol. 12, no. 6, p. e0177678, Jun. 2017, doi: 10.1371/journal.pone.0177678.

[31] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.