

Model Development Phase Template

Date	July 2024
Team ID	739656
Project Title	Zombie detection using the machine learning
Maximum Marks	10 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

Initial Model Training Code (5 marks):

Paste the screenshot of the model training code

Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics																														
Model 1	Logistic regression model typically include accuracy, precision, recall, F1 score to evaluate its predictive performance and generalization capability.	<div>LOGISTIC REGRESSION</div> <div><pre>[] #LOGISTIC REGRESSION #Importing the library from sklearn.linear_model import LogisticRegression #initializing the model lr=LogisticRegression() #fit the model lr.fit(x_train,y_train) #predict the model predic=lr.predict(x_test) #finding accuracy,classification report from sklearn.metrics import classification_report print(classification_report(y_test,predic))</pre></div> <div><table><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr><tr><td>0</td><td>0.65</td><td>0.82</td><td>0.73</td><td>1321</td></tr><tr><td>1</td><td>0.76</td><td>0.56</td><td>0.64</td><td>1305</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.69</td><td>2626</td></tr><tr><td>macro avg</td><td>0.71</td><td>0.69</td><td>0.69</td><td>2626</td></tr><tr><td>weighted avg</td><td>0.70</td><td>0.69</td><td>0.69</td><td>2626</td></tr></table></div>		precision	recall	f1-score	support	0	0.65	0.82	0.73	1321	1	0.76	0.56	0.64	1305	accuracy			0.69	2626	macro avg	0.71	0.69	0.69	2626	weighted avg	0.70	0.69	0.69	2626
	precision	recall	f1-score	support																												
0	0.65	0.82	0.73	1321																												
1	0.76	0.56	0.64	1305																												
accuracy			0.69	2626																												
macro avg	0.71	0.69	0.69	2626																												
weighted avg	0.70	0.69	0.69	2626																												
Model 2	Decision tree classifier model commonly include accuracy, precision, recall, F1 score which help assess the model's prediction accuracy and generalizability	<div><pre>#decision tree classifier from sklearn.tree import DecisionTreeClassifier dec=DecisionTreeClassifier() dec.fit(x_train,y_train) predi=dec.predict(x_test) from sklearn.metrics import classification_report print(classification_report(y_test,predi))</pre></div> <div><table><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr><tr><td>0</td><td>0.71</td><td>0.69</td><td>0.70</td><td>1321</td></tr><tr><td>1</td><td>0.69</td><td>0.71</td><td>0.70</td><td>1305</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.70</td><td>2626</td></tr><tr><td>macro avg</td><td>0.70</td><td>0.70</td><td>0.70</td><td>2626</td></tr><tr><td>weighted avg</td><td>0.70</td><td>0.70</td><td>0.70</td><td>2626</td></tr></table></div>		precision	recall	f1-score	support	0	0.71	0.69	0.70	1321	1	0.69	0.71	0.70	1305	accuracy			0.70	2626	macro avg	0.70	0.70	0.70	2626	weighted avg	0.70	0.70	0.70	2626
	precision	recall	f1-score	support																												
0	0.71	0.69	0.70	1321																												
1	0.69	0.71	0.70	1305																												
accuracy			0.70	2626																												
macro avg	0.70	0.70	0.70	2626																												
weighted avg	0.70	0.70	0.70	2626																												
Model 3	Random forest classifier model often encompass accuracy, precision, recall, F1 score to measure its prediction quality and robustness.	<div>RANDOM FOREST CLASSIFIER</div> <div><pre>[45] #random forest classifier from sklearn.ensemble import RandomForestClassifier rfc=RandomForestClassifier() rfc.fit(x_train,y_train) pred=rfc.predict(x_test) from sklearn.metrics import classification_report print(classification_report(y_test,pred))</pre></div> <div><table><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr><tr><td>0</td><td>0.69</td><td>0.87</td><td>0.77</td><td>1321</td></tr><tr><td>1</td><td>0.82</td><td>0.61</td><td>0.70</td><td>1305</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.74</td><td>2626</td></tr><tr><td>macro avg</td><td>0.76</td><td>0.74</td><td>0.73</td><td>2626</td></tr><tr><td>weighted avg</td><td>0.76</td><td>0.74</td><td>0.73</td><td>2626</td></tr></table></div>		precision	recall	f1-score	support	0	0.69	0.87	0.77	1321	1	0.82	0.61	0.70	1305	accuracy			0.74	2626	macro avg	0.76	0.74	0.73	2626	weighted avg	0.76	0.74	0.73	2626
	precision	recall	f1-score	support																												
0	0.69	0.87	0.77	1321																												
1	0.82	0.61	0.70	1305																												
accuracy			0.74	2626																												
macro avg	0.76	0.74	0.73	2626																												
weighted avg	0.76	0.74	0.73	2626																												

Model 4	K-nearest neighbors classifier model typically include accuracy, precision, recall, F1 score to evaluate its prediction performance and generalization ability	<div>K-NEAREST NEIGHBORS</div> <pre>[48] #knn from sklearn.neighbors import KNeighborsClassifier knn=KNeighborsClassifier() knn.fit(x_train,y_train) p=knn.predict(x_test) from sklearn.metrics import classification_report print(classification_report(y_test,p))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.69</td><td>0.79</td><td>0.74</td><td>1321</td></tr><tr><td>1</td><td>0.75</td><td>0.65</td><td>0.69</td><td>1305</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.72</td><td>2626</td></tr><tr><td>macro avg</td><td>0.72</td><td>0.72</td><td>0.72</td><td>2626</td></tr><tr><td>weighted avg</td><td>0.72</td><td>0.72</td><td>0.72</td><td>2626</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.69	0.79	0.74	1321	1	0.75	0.65	0.69	1305	accuracy			0.72	2626	macro avg	0.72	0.72	0.72	2626	weighted avg	0.72	0.72	0.72	2626
	precision	recall	f1-score	support																												
0	0.69	0.79	0.74	1321																												
1	0.75	0.65	0.69	1305																												
accuracy			0.72	2626																												
macro avg	0.72	0.72	0.72	2626																												
weighted avg	0.72	0.72	0.72	2626																												
Model 5	Ridge classifier model typically include accuracy, precision, recall, F1 score, and mean squared error to evaluate its prediction performance and generalization.	<div>RIDGE CLASSIFIER</div> <pre>[50] #RIDGE CLASSIFIER from sklearn.linear_model import RidgeClassifier rg=RidgeClassifier() rg.fit(x_train,y_train) p=rg.predict(x_test) from sklearn.metrics import classification_report print(classification_report(y_test,p))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.65</td><td>0.80</td><td>0.72</td><td>1321</td></tr><tr><td>1</td><td>0.74</td><td>0.56</td><td>0.64</td><td>1305</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.68</td><td>2626</td></tr><tr><td>macro avg</td><td>0.69</td><td>0.68</td><td>0.68</td><td>2626</td></tr><tr><td>weighted avg</td><td>0.69</td><td>0.68</td><td>0.68</td><td>2626</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.65	0.80	0.72	1321	1	0.74	0.56	0.64	1305	accuracy			0.68	2626	macro avg	0.69	0.68	0.68	2626	weighted avg	0.69	0.68	0.68	2626
	precision	recall	f1-score	support																												
0	0.65	0.80	0.72	1321																												
1	0.74	0.56	0.64	1305																												
accuracy			0.68	2626																												
macro avg	0.69	0.68	0.68	2626																												
weighted avg	0.69	0.68	0.68	2626																												