Practical Data Science

COSC2789 - Dr Thuy Nguyen

Teams 2:

Tran Phan Hoang Phuc (s3929597)
Bui Minh Nhat (s3878174)
Nguyen Xuan Thanh (s3915468)

RMIT UNIVERSITY

I. Introduction

Introduction

In **2022**, it is estimated that more than **18.1 million** cases of cancer are recorded worldwide [1]. Cancer prevention is one of the **most crucial and foremost** public health concerns of modern society, due to its **rapid growth rate and limitation in cure**.

Introduction

Colorectal Cancer, or ‘Colon’ cancer for short, is a disease where **cells within the rectum or colon** of a human body **multiplies and grows at an uncontrollable rate** [8]. It is one of the top three most popular forms of cancer worldwide [9].

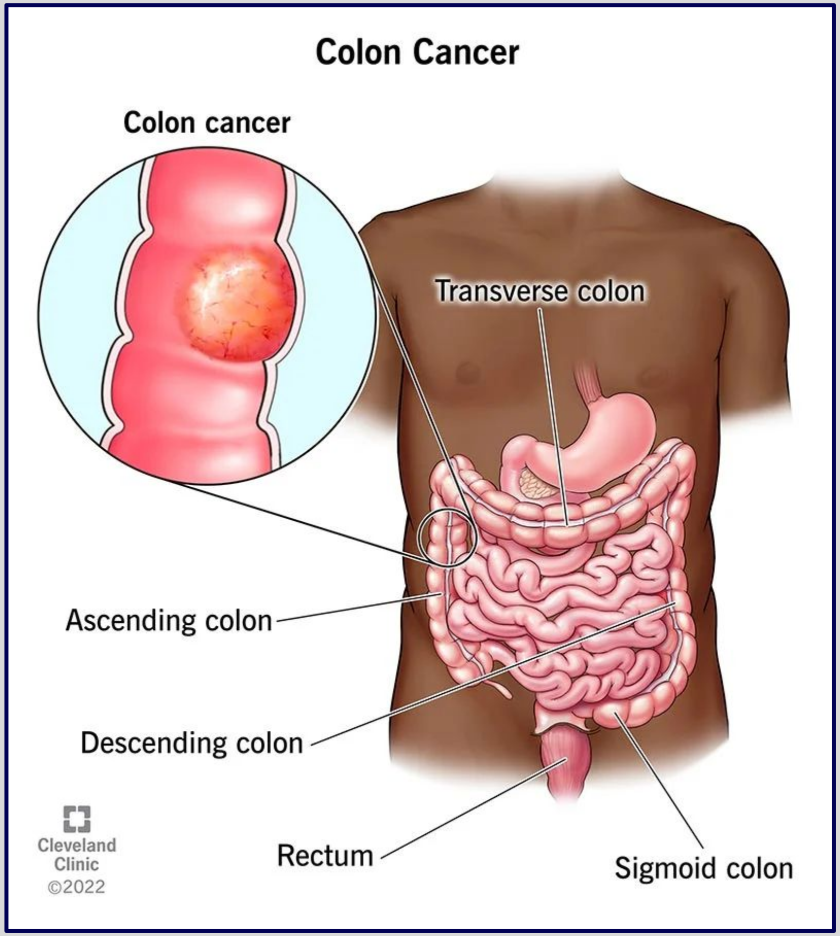
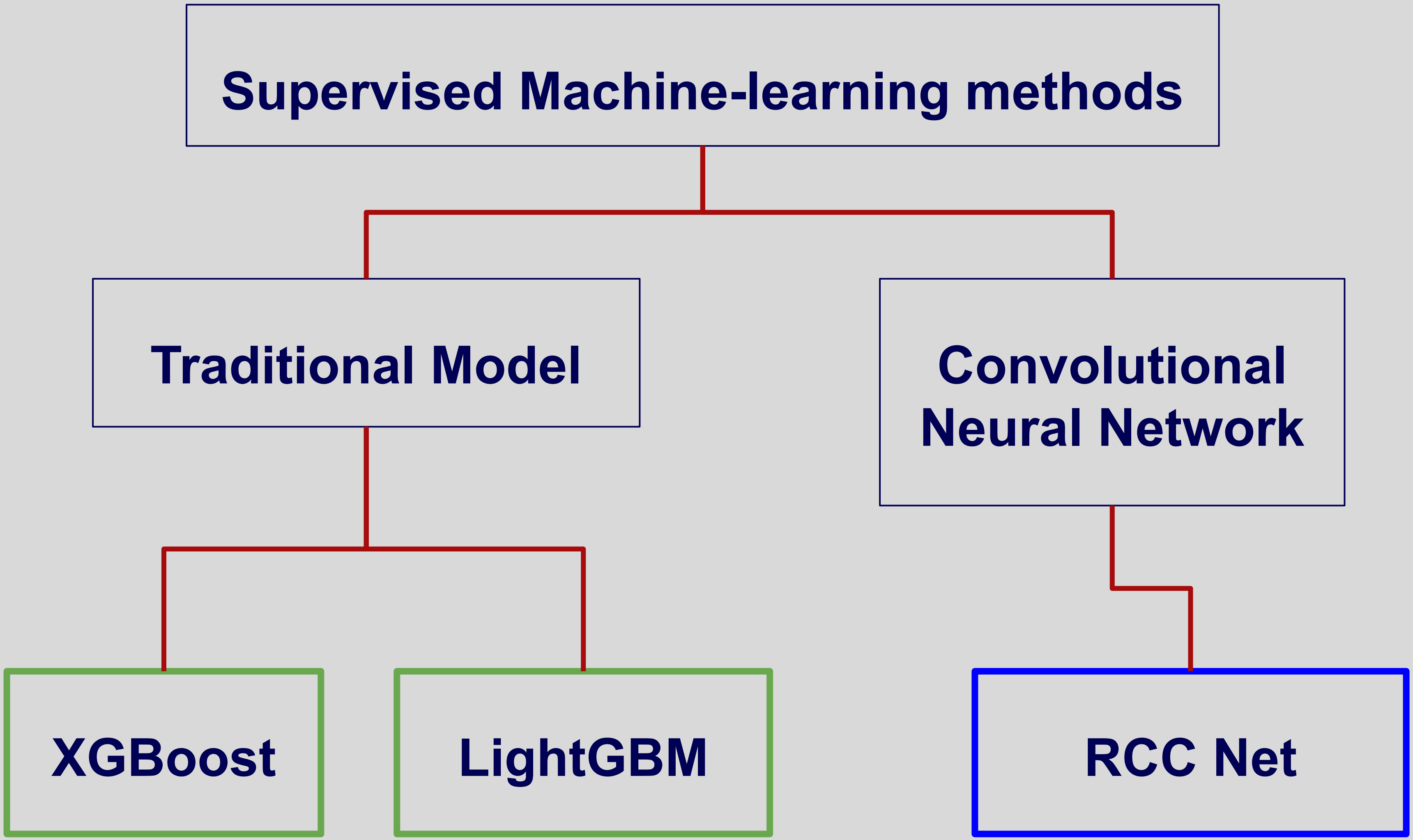


Fig 1.1: Colon cancer anatomy layout

Introduction



Dataset (Main data)

6 Columns &
9896 Columns

4 Types: firoblase, inflammatory, epithelial, others

data_labels_mainData

InstanceID	patientID	ImageName	cellTypeName	cellType	isCancerous
22405	1	22405.png	fibroblast	0	0
22406	1	22406.png	fibroblast	0	0
22407	1	22407.png	fibroblast	0	0
22408	1	22408.png	fibroblast	0	0
22409	1	22409.png	fibroblast	0	0
22410	1	22410.png	fibroblast	0	0

Row ID

Fig 1.2: first 6 rows of main data set [2]

Unique to each patient

0 (false)
1 (true)

Dataset (Extra)

4 Columns & 10384 Rows

data_labels_extraData

InstanceID	patientID	ImageName	isCancerous
12681	61	12681.png	0
12682	61	12682.png	0
12683	61	12683.png	0
12684	61	12684.png	0
12685	61	12685.png	0

Fig 1.3: first 5 rows of extra data set [2]

Row ID

0 (false)
1 (true)

Unique to each patient

II. Methodology

RMIT UNIVERSITY

Data pipeline

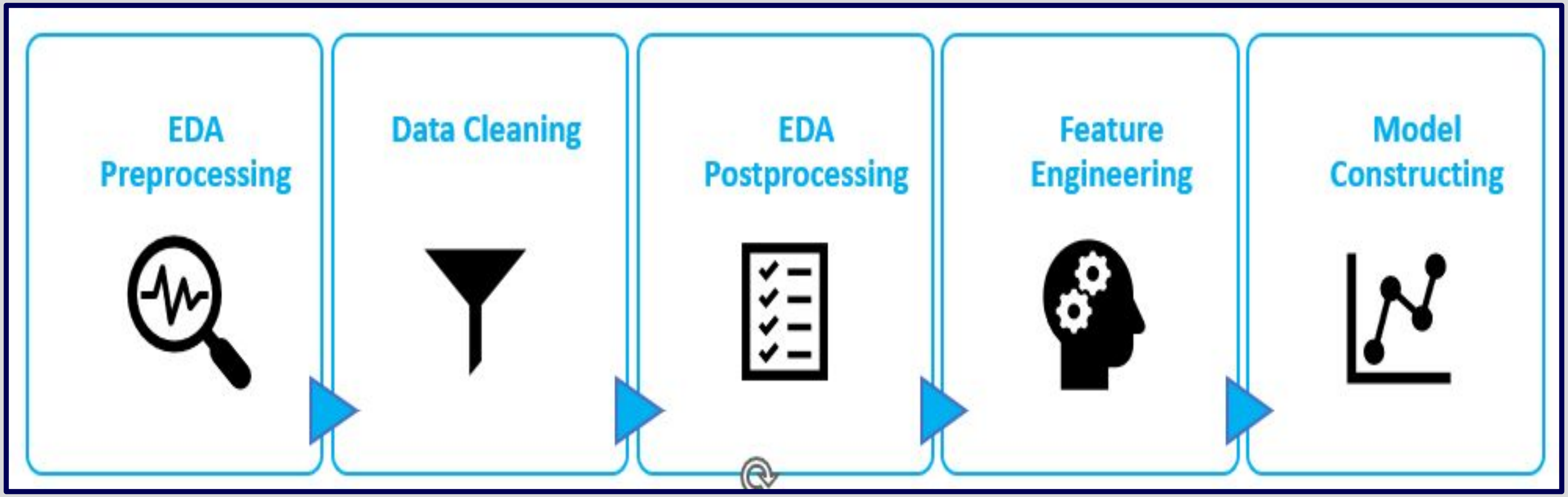


Fig 2.1: Data Pipeline visualization

Traditional Machine Learning models

RMIT UNIVERSITY

XGBOOST Explanation

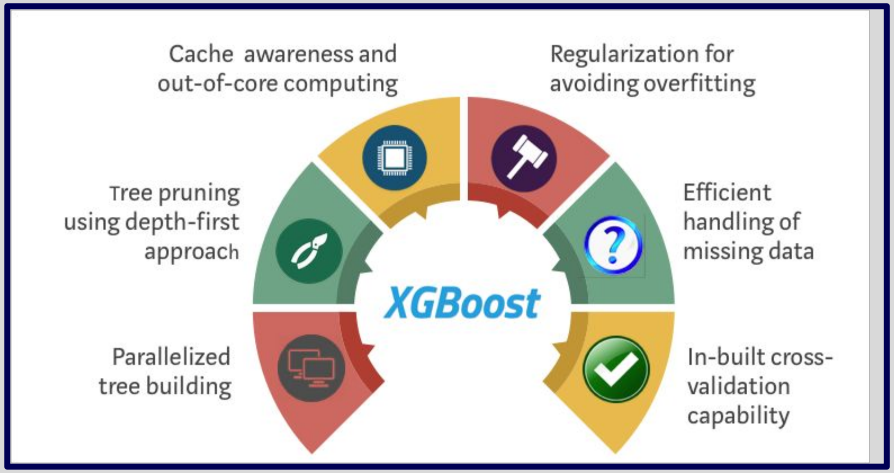


Fig 3.1: XGBoost features visualization

Extreme Gradient Boosting (XGBoost) is a **distributed, scalable gradient-boosted decision tree** (GBDT) machine learning framework.

It's consider to be one of the top machine learning library for **regression, classification, and ranking issues**, it offers parallel tree boosting [3].

XGBOOST

The XGBOOST Library is the implementation of Gradient Boosting algorithm.

Boosting is a modeling approach combining several weak tree based classifiers, with aims to create a high performance classifier.

In gradient boosting, **each classifiers correct it's predecessor error**. Additionally each classifier is trained using the residual errors of predecessor as labels

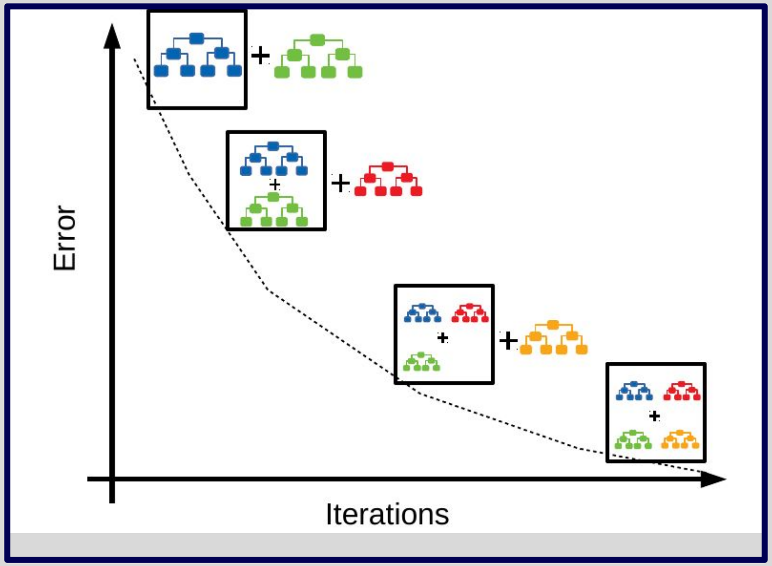


Fig 3.2: ___

LightGBM

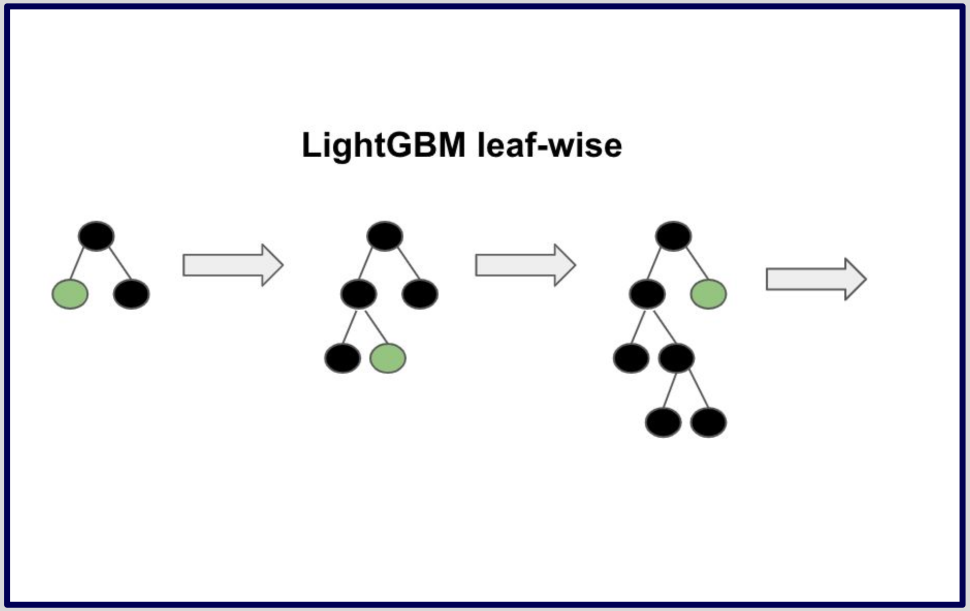


Fig 4.1: LightGBM leaf-wise grow

LightGBM (short for Light Gradient Boosting Machine), is a framework that use **tree-based learning algorithms** [4]**.** It grows the decision tree **leaf-wise** (vertically) which differs than other boosting algorithms where they grows depth-wise (horizontally) [5].

LightGBM

It can handle large datasets and is **faster** than other popular libraries (ex: XGBoost).

LightGBM includes unique features [10]:

+ Support for categorical features

+ Efficient handling of missing values

+ Support for parallel processing.

Convolutional Neural Network

Convolutional Neural Network

CNN (short for Convolutional Neural Network) is a type of **deep learning model** for processing data that has a **grid pattern**, such as images [6].

It can deal with **2D image classification** problems with **higher accuracy.**

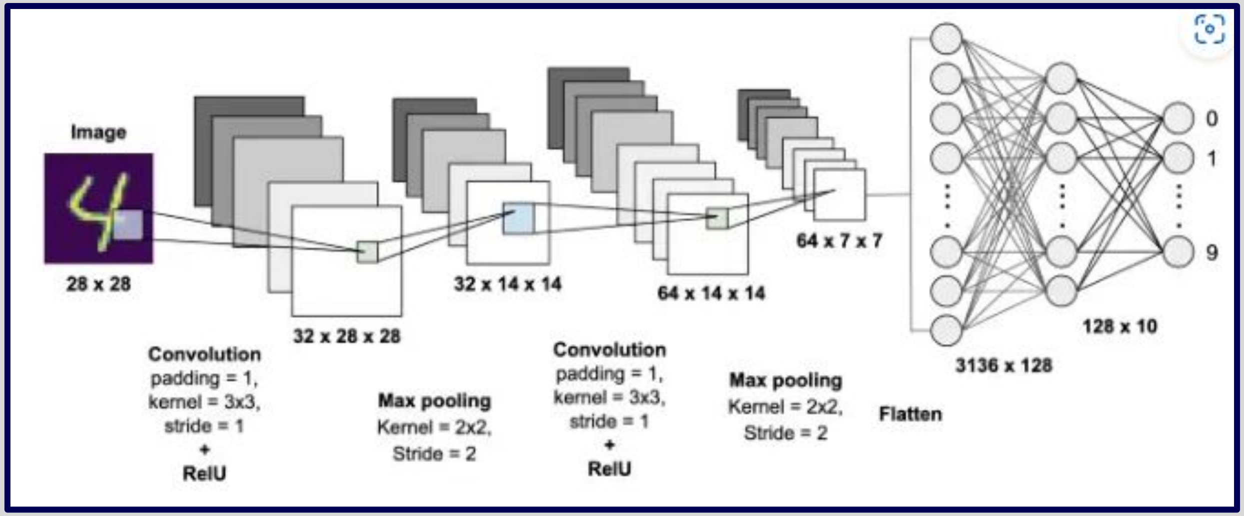


Fig 5.1: CNN deep learning visualization

RCC NET

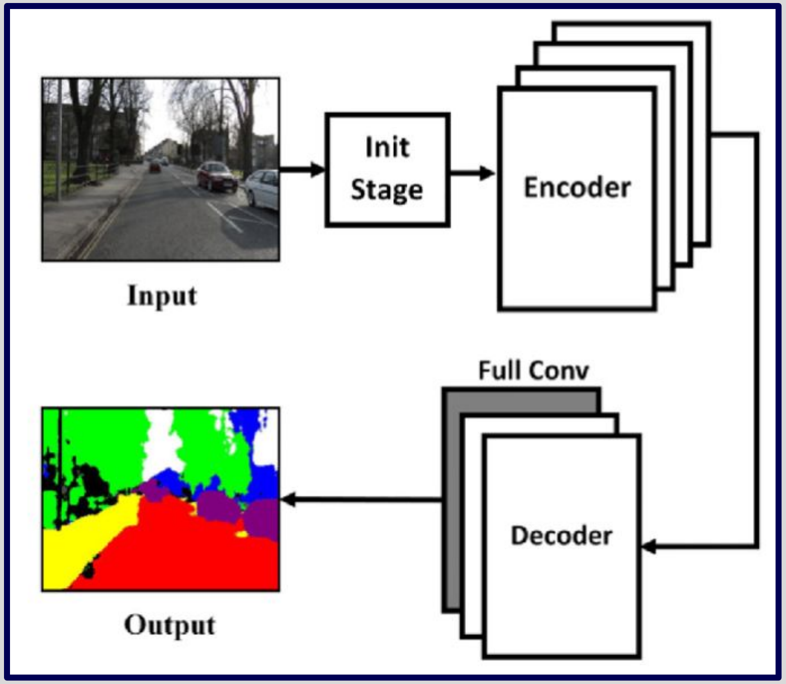


Fig 5.2: ____

RCC Net is a deep learning model (CNN based architecture).

RCC-Net performs dimensionality reduction to compress and extract relevant feature.

ENSEMBLE MODEL

Our **ensemble classifier** of choice is a **VotingClassifier** provided in Scikit learn library.

Inspired by the concept of Boosting, the team wanted to create a new model based on the combination of high performances.

The submodels a chance to “vote” the probability of the output result. The weights of the “vote” can be modified [7].

ENSEMBLE MODEL

The model start with a **scaler** (MinMax scaler).

The post scaling data then pass through the **Voting Classifier** ensemble model consisting of a **tuned LightGBM** classifier and **XGBoost Classifier** for training and classification.

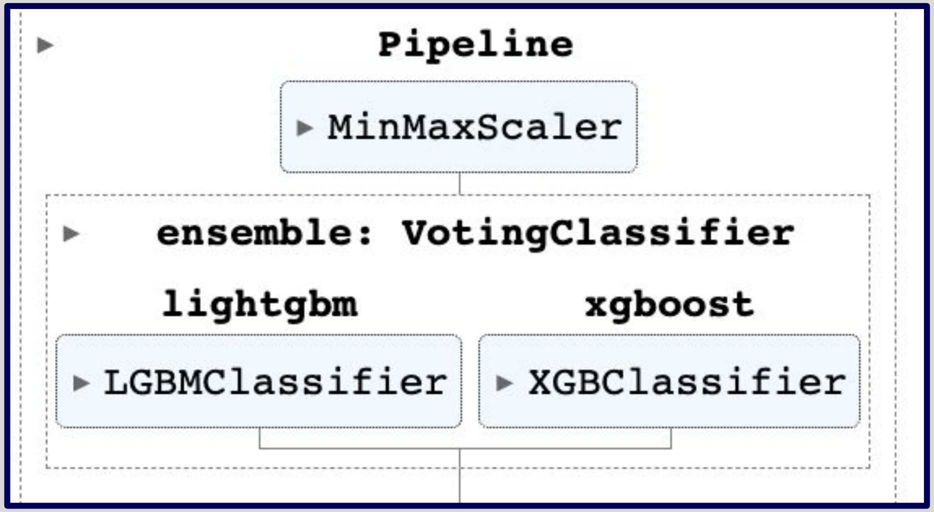


Fig 6.1: Ensemble model visualization

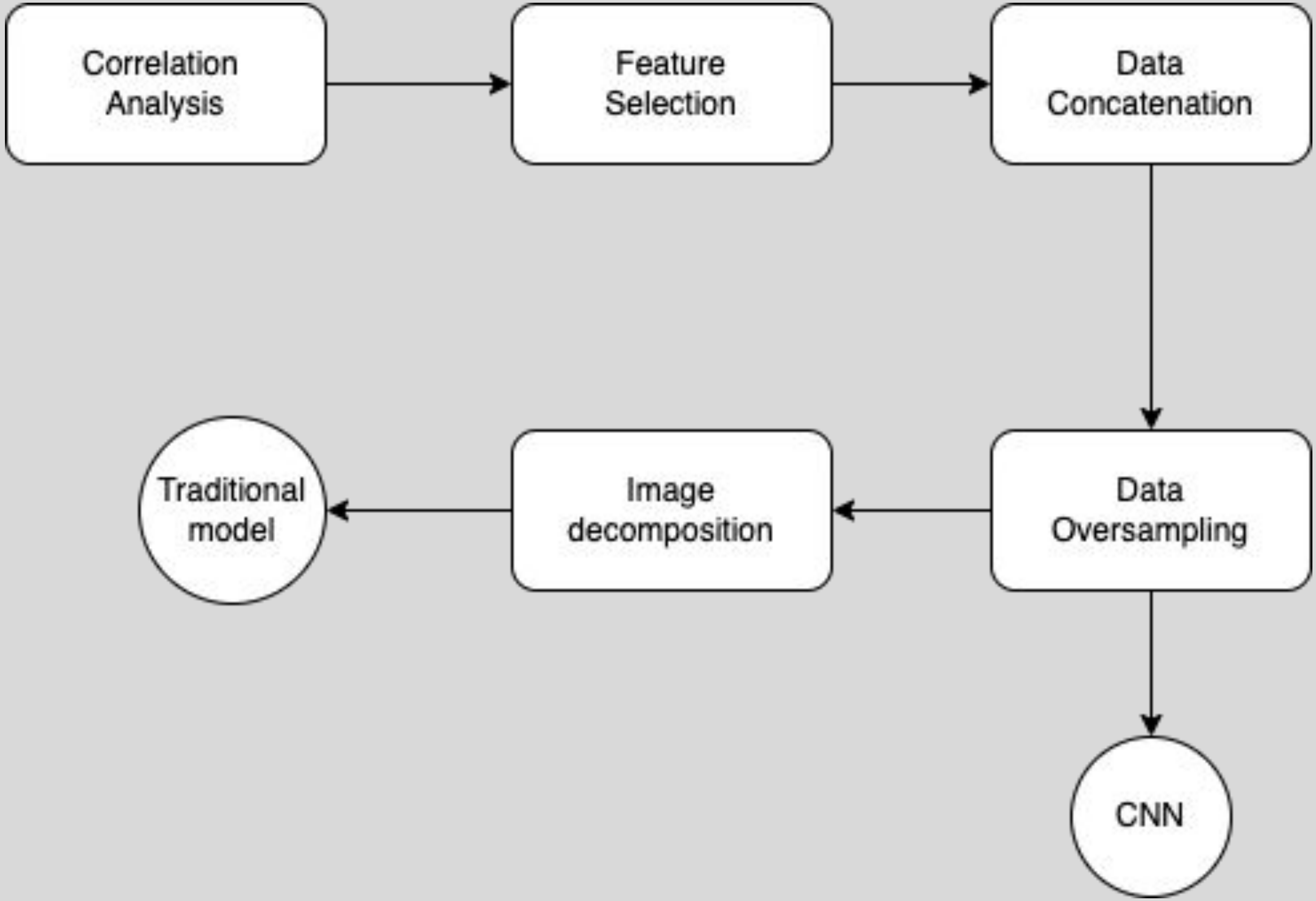
III. Feature Engineering

EDA [Pre & Post Processing]

Hypotheses to be examined (apply for both **before** and **after** processing the data):

- **Hypothesis 1:** Cell type 2 is the only cancerous cell
- **Hypothesis 2:** Cell type 2 have the highest frequency
- **Hypothesis 3:** Cancerous cells account for over 50%
- **Hypothesis 4:** The percentage of images being identified as cancerous for each patientID will be over 50%

Feature Engineering



Feature Engineering

Main data set:
'celltype' and 'patientid' are correlated with the 'iscancerous' column.

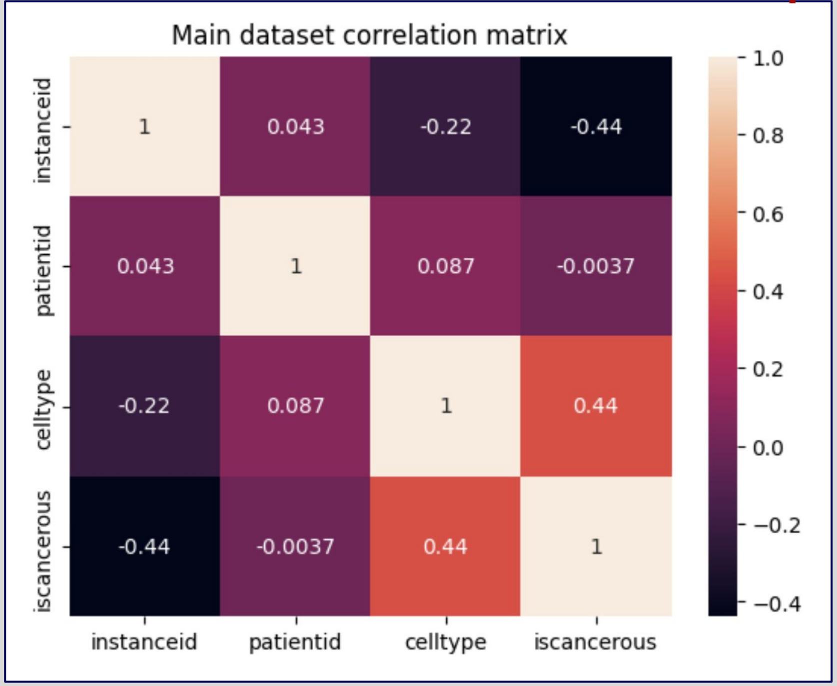


Fig 7.1: Main Data correlation matrix

Extra data set:
'patientid' and 'instanceid' are correlated with the 'iscancerous' column.

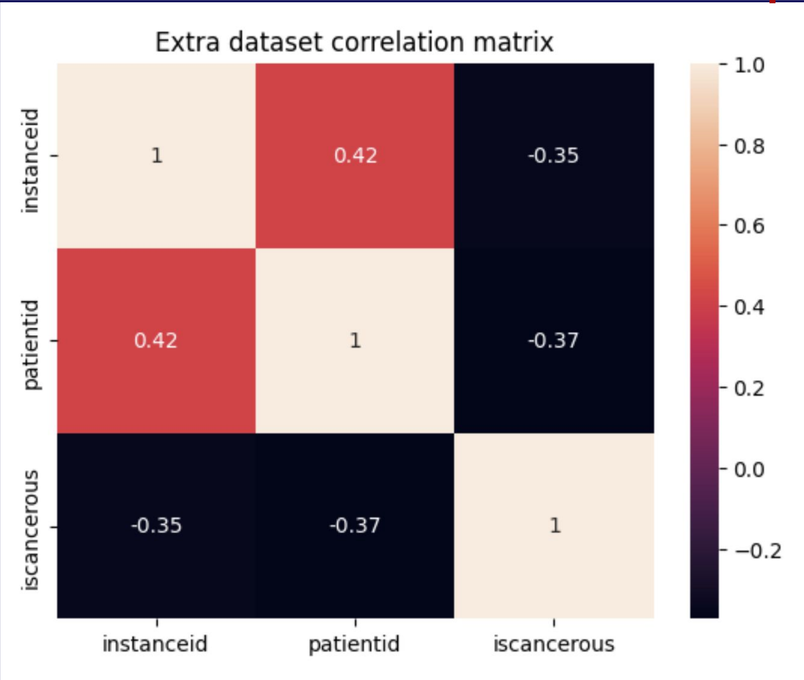


Fig 7.2: Extra Data correlation matrix

Feature Engineering

Main data set:
'imagename', 'iscancerous' and 'celltype' is selected,
The remaining columns is dropped to prevent data leakage.

```
Data columns (total 3 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   imagename    9803 non-null   object
 1   celltype     9803 non-null   int64
 2   iscancerous  9803 non-null   int64
dtypes: int64(2), object(1)
memory usage: 229.9+ KB
```

Fig 7.3: Main data set columns selection for modeling

Extra data set:
'iscancerous' and 'imagename' is selected.

```
Int64Index: 20097 entries, 0 to 20096
Data columns (total 2 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   imagename    20097 non-null  object
 1   iscancerous  20097 non-null  int64
dtypes: int64(1), object(1)
memory usage: 471.0+ KB
```

Fig 7.4: Extra data set columns selection for modeling

Data Concatenation

Both extra and main dataset have ‘iscancerous’, and ‘imagename’

=> A **new dataset is created** by **concatenate** two feature of the two dataset:

- In order to increase the size of a dataset
- Enhancing the performance of the model

Over Sampling

Using **SMOTE** from imbalanced learn library. The dataset is for **task 1**.

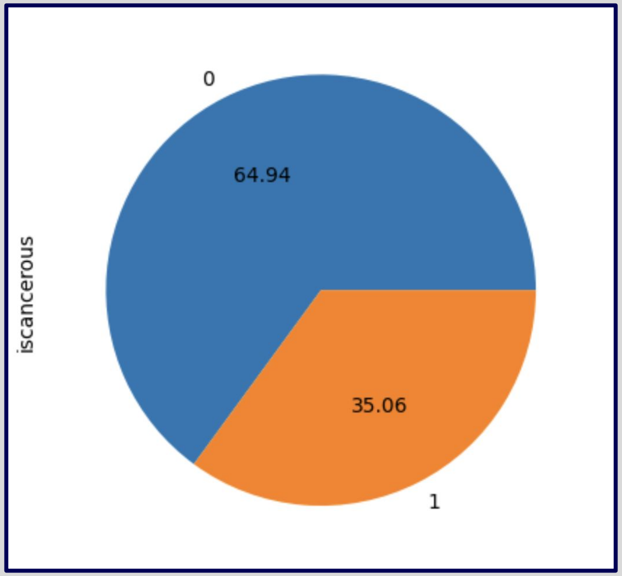


Fig 7.5: Before over sampling for task 1

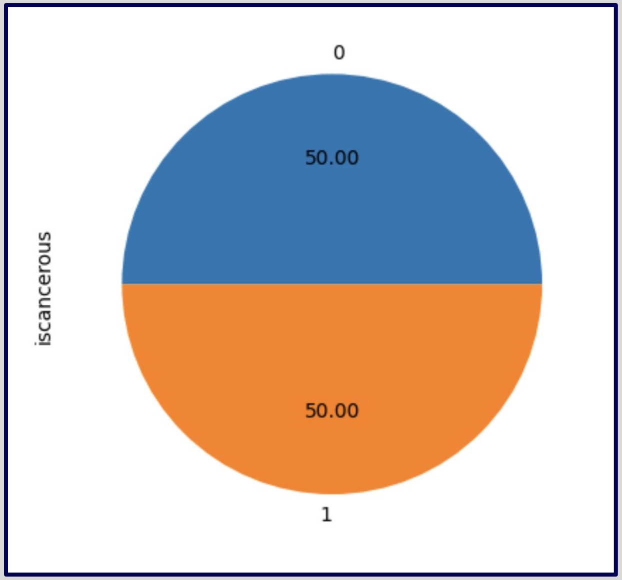


Fig 7.6: After over sampling for task 1

Over Sampling

Using **SMOTE** from imbalanced learn library. The dataset is for **task 2**.

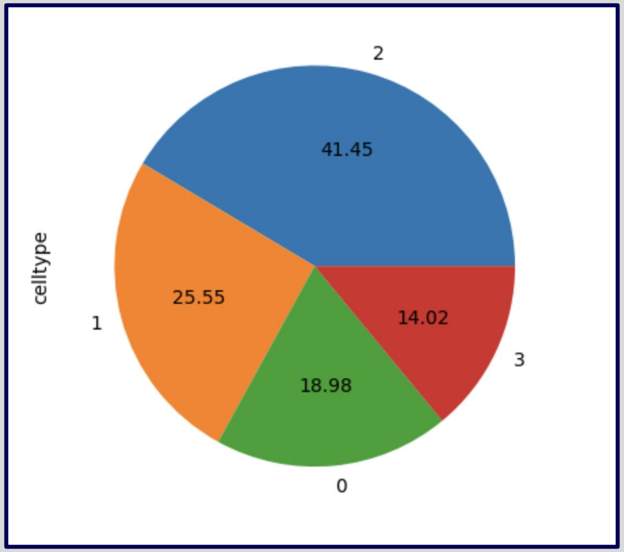


Fig 7.7: Before over sampling for task 2

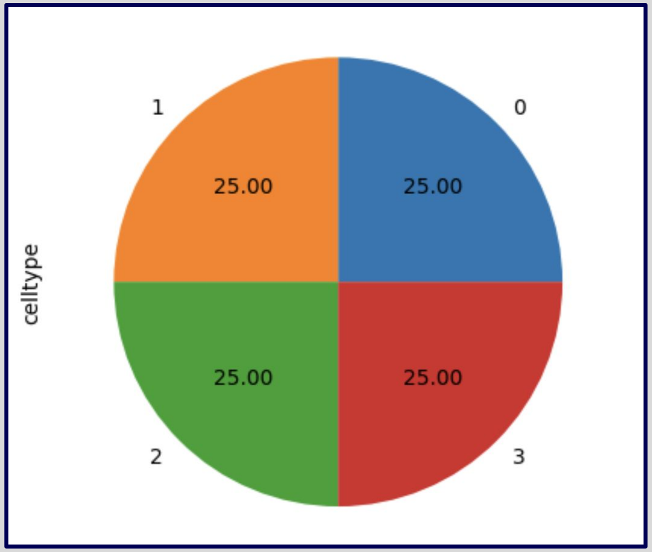


Fig 7.8: After over sampling for task 2

Image Decomposition

Images in **raw format** are **not compatible** with traditional machine learning framework.

- The 27x 27 x 3 size image is unravel into pixel with numerical features.

Fig 8.1: Example of a cancer cell image

Visualize

iscancerous int64 0 - 1	p0 uint8 38 - 255	p1 uint8 14 - 255	p2 uint8	p3 uint8	p4 uint8
0	198	151	196	204	158
0	220	193	228	216	189
0	255	226	249	255	234
0	231	191	233	196	152
0	212	179	220	199	164
0	191	138	190	203	151

Fig 8.2: Numerical features from pixel image

IV. Results

RMIT UNIVERSITY

Tuned XGBOOST results

Task 1: Tuned XGBooost

```
Model accuracy for train set: 1.000
Model accuracy for test set: 0.926

              precision    recall  f1-score   support

           0       0.94      0.91      0.92      3915
           1       0.91      0.94      0.93      3915

    accuracy                           0.93      7830
   macro avg       0.93      0.93      0.93      7830
weighted avg       0.93      0.93      0.93      7830
```

Fig 9.1: XGBoost accuracy score (task 1)

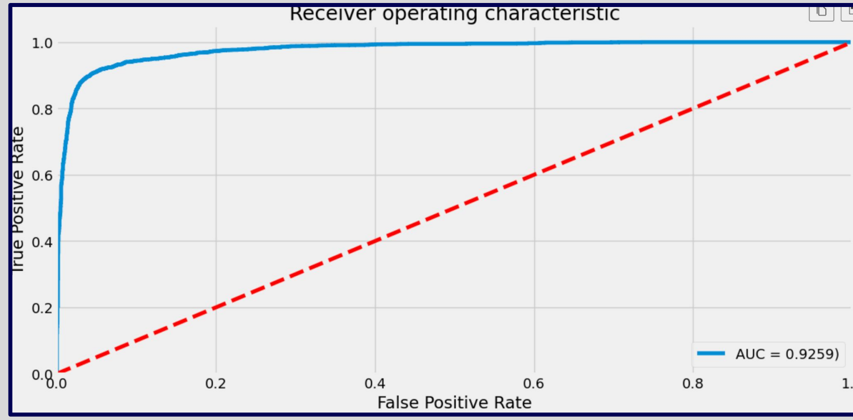


Fig 9.2: Tuned XGBoost ROC Curve (task 1)

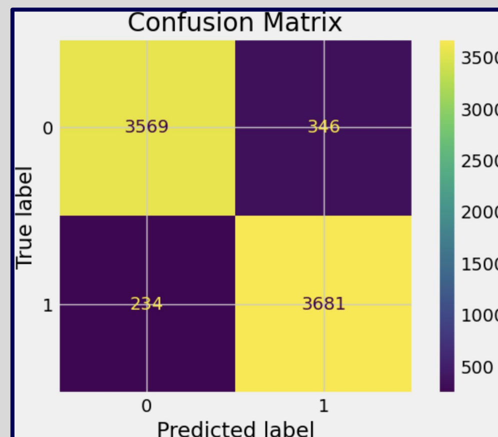


Fig 9.3: XGBoost Confusion Matrix (task 1)

Tuned XGBOOST results

Task 2: Tuned XGBooost

```
Model accuracy for train set: 1.000
Model accuracy for test set: 0.911
```

	precision	recall	f1-score	support
0	0.90	0.95	0.93	1219
1	0.87	0.89	0.88	1219
2	0.92	0.89	0.91	1219
3	0.96	0.91	0.93	1219
accuracy			0.91	4876
macro avg	0.91	0.91	0.91	4876
weighted avg	0.91	0.91	0.91	4876

Fig 9.4: XGBoost accuracy score (task 2)

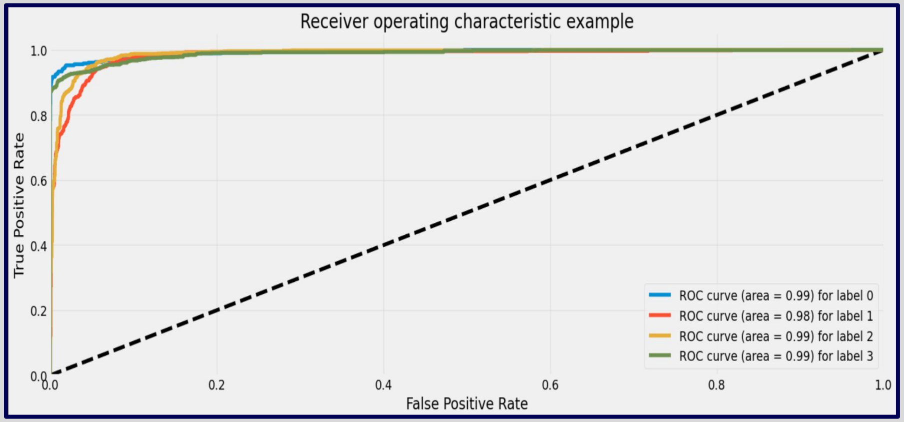


Fig 9.5: XGBoost ROC Curve (task 2)

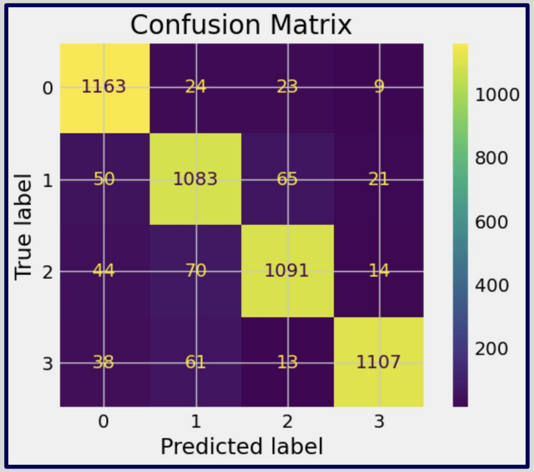


Fig 9.6: XGBoost confusion matrix (task 2)

Tuned LightGBM results

Task 1: Tuned LightGBM

```
Model accuracy for train set: 0.979
Model accuracy for test set: 0.912

              precision    recall  f1-score   support

           0       0.92      0.90      0.91      3915
           1       0.90      0.93      0.91      3915

    accuracy                           0.91      7830
   macro avg       0.91      0.91      0.91      7830
weighted avg       0.91      0.91      0.91      7830
```

Fig 9.7: Tuned LightGBM accuracy score (task 1)

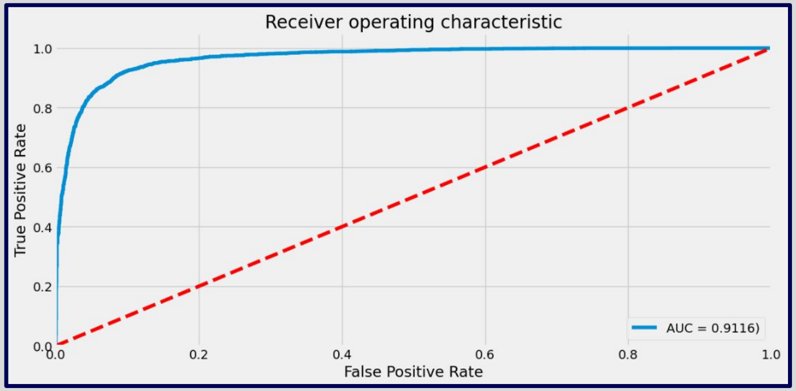


Fig 9.8: Tuned LightGBM ROC curve (task 1)

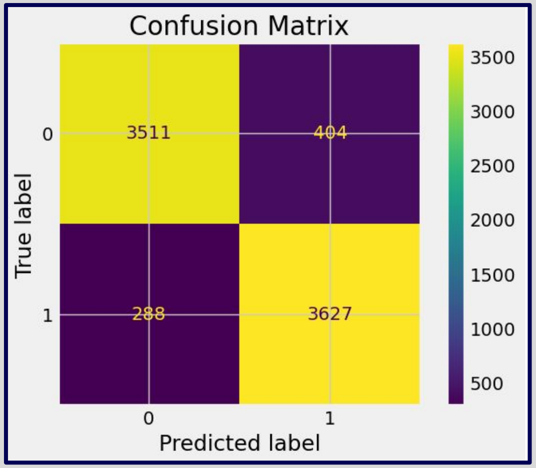


Fig 9.9: Tuned LightGBM confusion matrix (task 1)

Tuned LightGBM results

Task 2: Tuned LightGBM

```
Model accuracy for train set: 1.000
Model accuracy for test set: 0.914

              precision    recall  f1-score   support

           0       0.91      0.96      0.93      1219
           1       0.88      0.89      0.89      1219
           2       0.92      0.90      0.91      1219
           3       0.96      0.90      0.93      1219

    accuracy                           0.91      4876
   macro avg       0.91      0.91      0.91      4876
weighted avg       0.91      0.91      0.91      4876


Confusion Matrix:
 [[1169   19   17   14]
 [  44 1089   61   25]
 [  44   68 1095   12]
 [  34   66   17 1102]]

Accuracy Score:  0.914
```

Fig 9.10: Tuned LightGBM accuracy score (task 2)

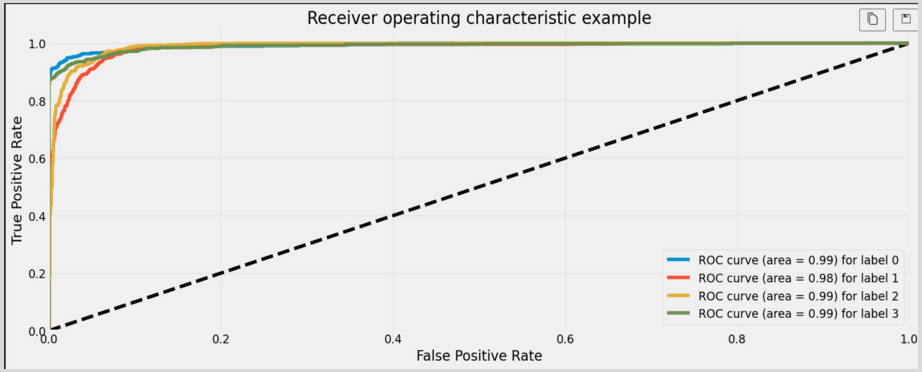


Fig 9.11: Tuned LIghtGBM ROC Curve (task 2)

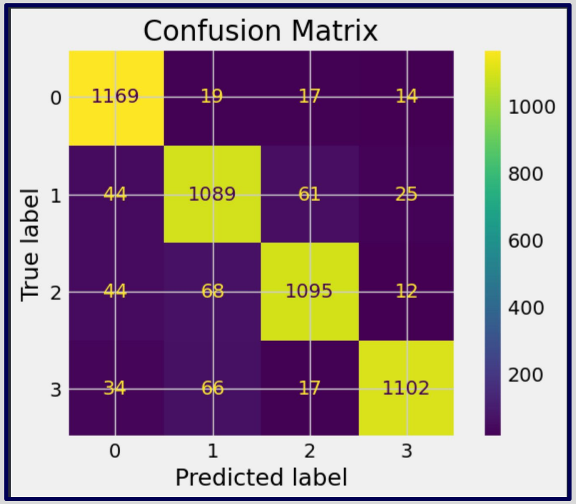


Fig 9.12: Tuned LightGBM confusion matrix (task 2)

Tuned RCC-net results

Task 1: Tuned RCC-net

```
Model Evaluation
---------------------------------------------
loss - test: 0.18401867151260376
loss - train : 0.09642554074525833
loss - val: 0.1682843416929245
---------------------------------------------
accuracy - test: 0.9329643249511719
accuracy - train : 0.9667692184448242
accuracy - val: 0.9483076930046082
---------------------------------------------
recall - test: 0.9310897588729858
recall - train : 0.9657047986984253
recall - val: 0.9473039507865906
---------------------------------------------
precision - test: 0.9324603080749512
precision - train : 0.967363715171814
precision - val: 0.9491523504257202
---------------------------------------------
f1 - test: 0.9317014217376709
f1 - train : 0.9664754271507263
f1 - val: 0.9481966495513916
```

Fig 9.13: Tuned RCC_net accuracy score (task 1)

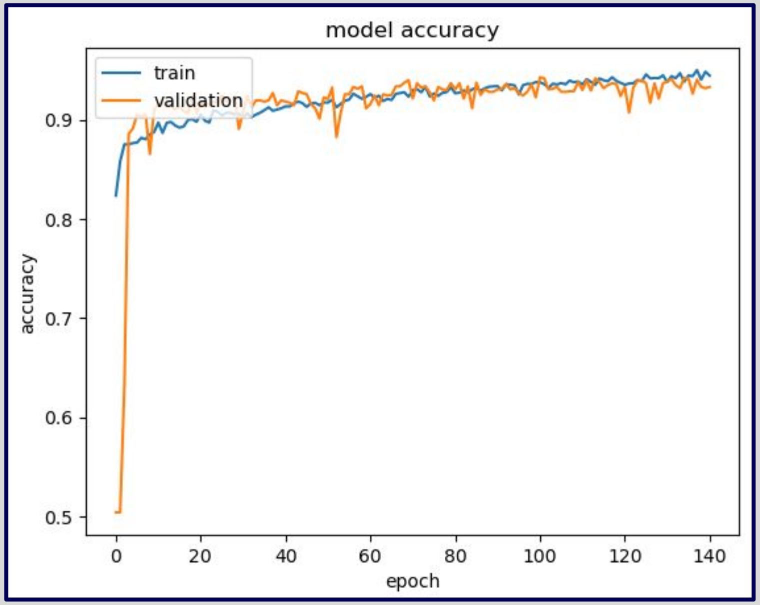


Fig 9.14: Tuned RCC_net model accuracy (task 1)

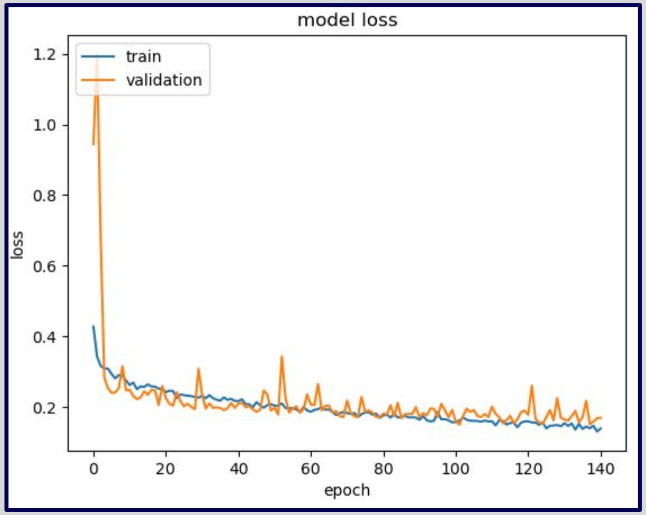


Fig 9.15: Tuned RCC_net model loss (task 1)

Tuned RCC-net results

Task 2: Tuned RCC-net

```
Model Evaluation
---------------------------------------------
loss - test: 0.3998308479785919
loss - train : 0.3382495045661926
loss - val: 0.3680887520313263
---------------------------------------------
accuracy - test: 0.8603506684303284
accuracy - train : 0.8728204965591431
accuracy - val: 0.876038134098053
---------------------------------------------
recall - test: 0.8403637409210205
recall - train : 0.8450912833213806
recall - val: 0.8542472720146179
---------------------------------------------
precision - test: 0.8802827596664429
precision - train : 0.8960369229316711
precision - val: 0.8930943012237549
---------------------------------------------
f1 - test: 0.8595331311225891
f1 - train : 0.869326114654541
f1 - val: 0.8728362917900085
```

Fig 9.16: Tuned RCC_net accuracy score (task 2)

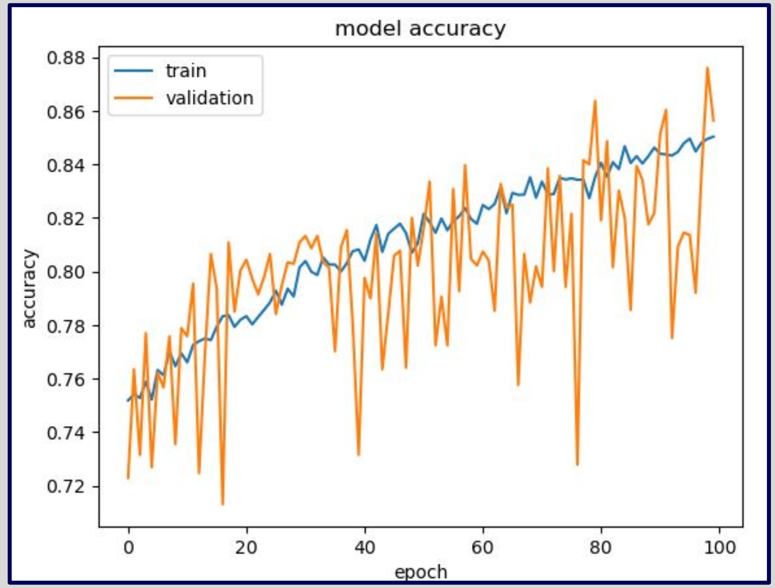


Fig 9.17: Tuned RCC_net model accuracy (task 2)

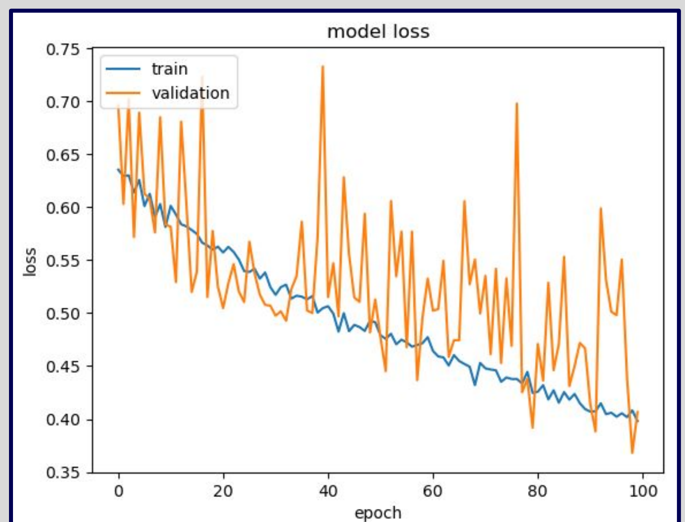


Fig 9.18: Tuned RCC_net model loss (task 2)

Ensemble Model

Task 1: Tuned Ensemble Model

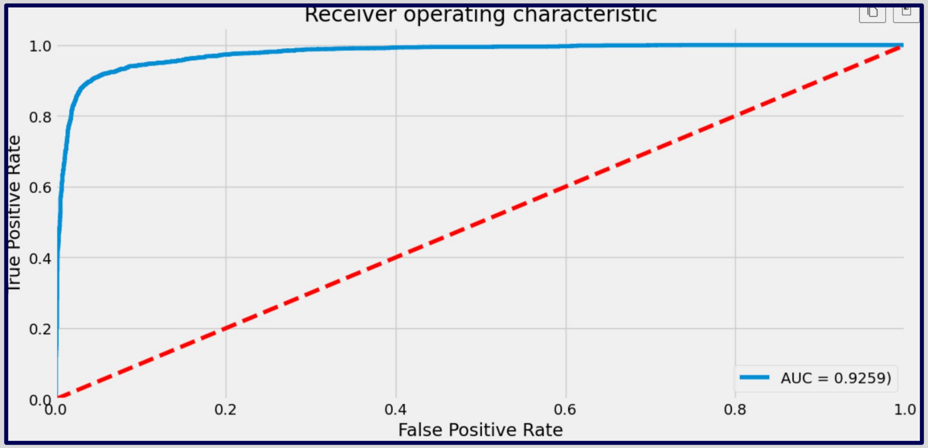


Fig 9.20: Tuned Ensemble ROC Curve (task 1)

Model accuracy for train set: 1.000
Model accuracy for test set: 0.926

	precision	recall	f1-score	support
0	0.94	0.91	0.92	3915
1	0.91	0.94	0.93	3915
accuracy			0.93	7830
macro avg	0.93	0.93	0.93	7830
weighted avg	0.93	0.93	0.93	7830

Fig 9.19: Tuned Ensemble accuracy score (task 1)

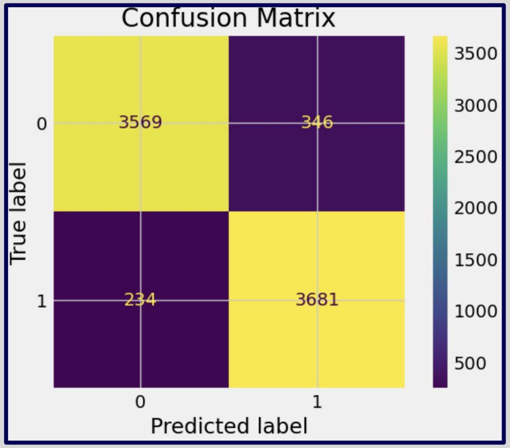


Fig 9.21: Tuned Ensemble model confusion matrix (task 1)

Ensemble Model

Task 2: Tuned Ensemble Model

```
Model accuracy for train set: 1.000
Model accuracy for test set: 0.911

              precision    recall  f1-score   support

           0       0.90      0.95      0.93      1219
           1       0.87      0.89      0.88      1219
           2       0.92      0.89      0.91      1219
           3       0.96      0.91      0.93      1219

    accuracy                           0.91      4876
   macro avg       0.91      0.91      0.91      4876
weighted avg       0.91      0.91      0.91      4876
```

Fig 9.22: Tuned Ensemble accuracy score (task 2)

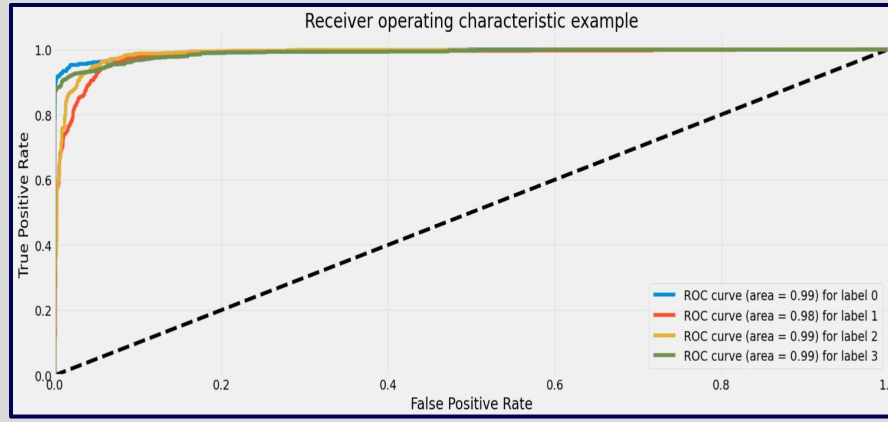


Fig 9.23: Tuned Ensemble ROC Curve (task 2)

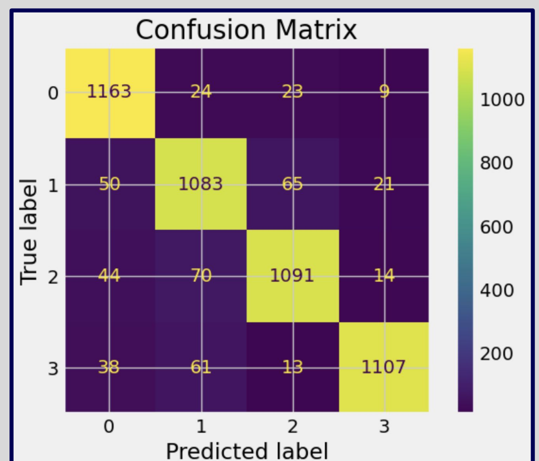
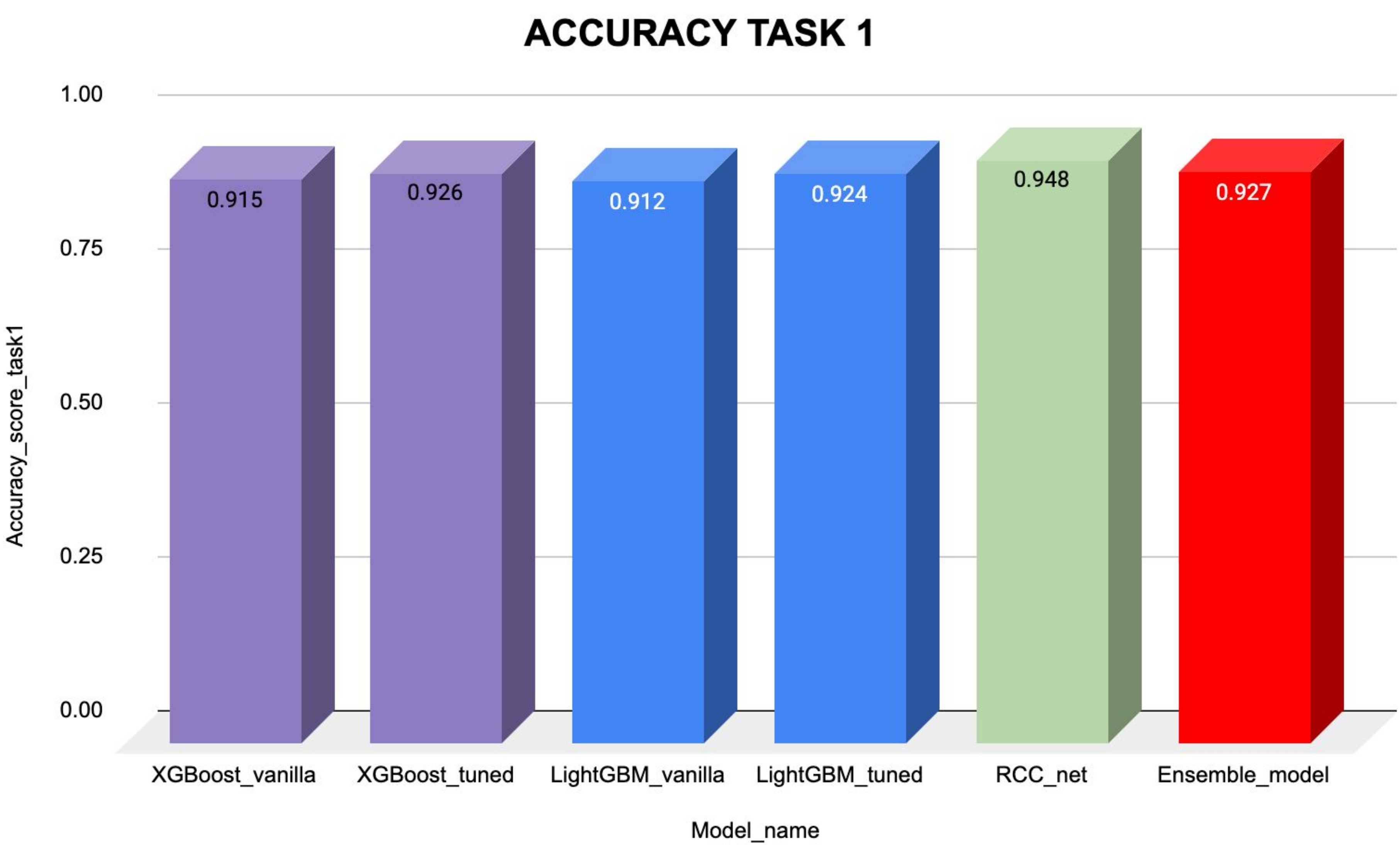


Fig 9.24: Tuned Ensemble model confusion matrix (task 2)

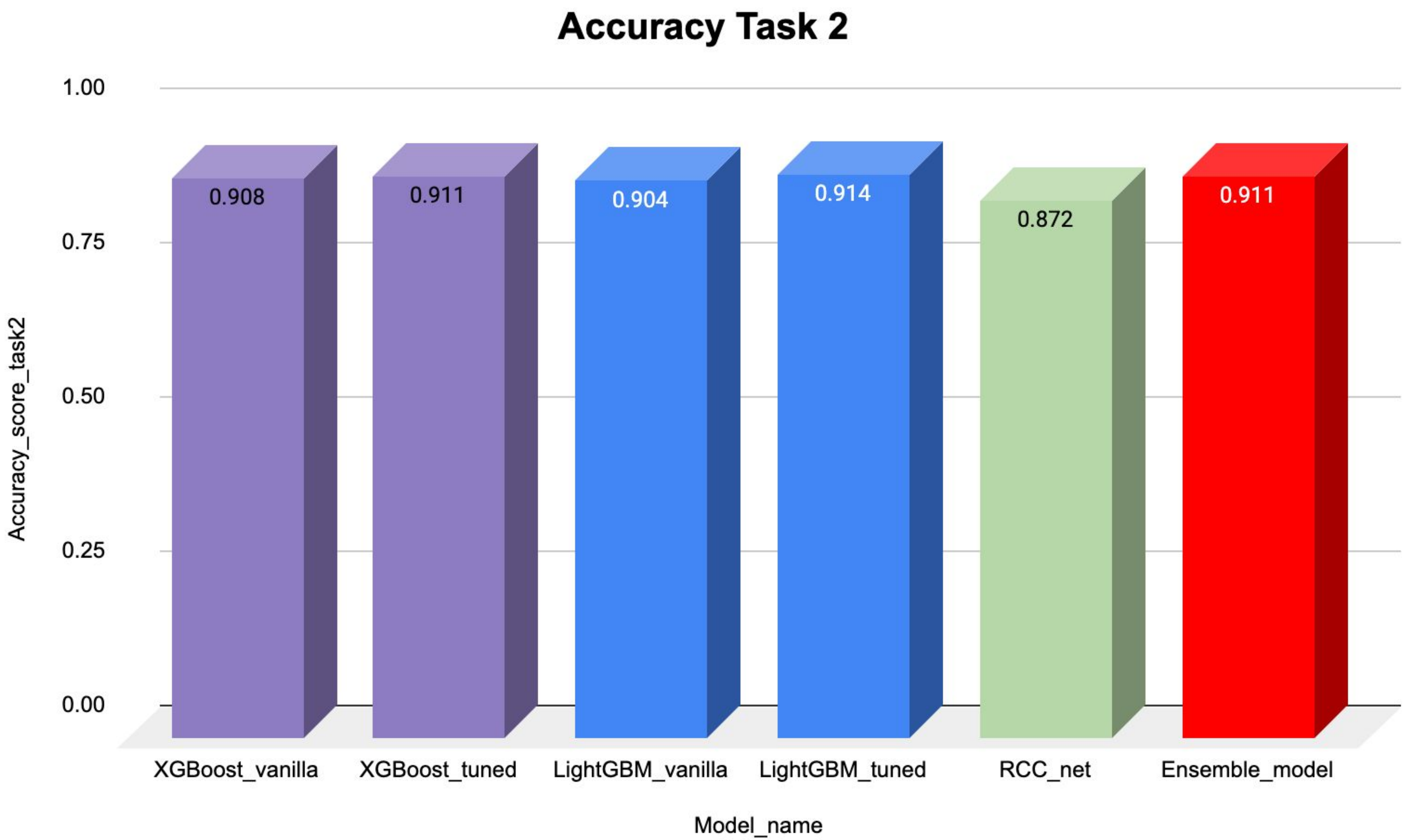
V. Discussion

RMIT UNIVERSITY

Result Model (Task 1)



Result Model (Task 2)



VI. Conclusion

RMIT UNIVERSITY

Conclusion

Best accuracy score for task 1: **RCC_net**

Score: 0.948

Best accuracy score for task 2: **LightGBM_tuned**

Score: 0.914

References

[1] World Cancer Research Fund International, “Worldwide cancer data: World cancer research fund international,” *WCRF International*, 14-Apr-2022. [Online]. Available: https://www.wcrf.org/cancer-trends/worldwide-cancer-data/. [Accessed: 17-Jan-2023].

[2] K. Sirinukunwattana, S. E. A. Raza, Y. Tsang, D. R. J. Snead, I. A. Cree and N. M. Rajpoot, "Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images," in IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1196

[3] Nvidia, “What is XGBoost?,” NVIDIA Data Science Glossary. [Online]. Available: https://www.nvidia.com/en-us/glossary/data-science/xgboost/. [Accessed: 17-Jan-2023].

[4] S. Shisingh, “Lightgbm (Light Gradient Boosting Machine),” GeeksforGeeks, 22-Dec-2021. [Online]. Available: https://www.geeksforgeeks.org/lightgbm-light-gradient-boosting-machine/. [Accessed: 17-Jan-2023].

[5] S. Surana, “What is light GBM? advantages & disadvantages? Light GBM vs XGBoost?: Data Science and Machine Learning,” Kaggle, 2021. [Online]. Available: https://www.kaggle.com/general/264327. [Accessed: 17-Jan-2023].

References

[6] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional Neural Networks: An overview and application in radiology - insights into imaging,” SpringerOpen, 22-Jun-2018. [Online]. Available: https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9#:~:text=CNN%20is%20a%20type%20of,%2D%20to%20high%2Dlevel%20patterns. [Accessed: 17-Jan-2023].

[7] S. Kumar, “Use voting classifier to improve the performance of your ML model,” *Medium*, 01-Nov-2021. [Online]. Available: https://towardsdatascience.com/use-voting-classifier-to-improve-the-performance-of-your-ml-model-805345f9de0e. [Accessed: 17-Jan-2023].

[8] “What is colorectal cancer?,” Centers for Disease Control and Prevention, 17-Feb-2022. [Online]. Available: https://www.cdc.gov/cancer/colorectal/basic_info/what-is-colorectal-cancer.htm#:~:text=Colorectal%20cancer%20is%20a%20disease,the%20colon%20to%20the%20anus. [Accessed: 17-Jan-2023].

[9] *Worldwide cancer data: World cancer research fund international*. WCRF International. (2022, April 14). Retrieved January 17, 2023, from https://www.wcrf.org/cancer-trends/worldwide-cancer-data/#:~:text=Breast%20and%20lung%20cancers%20were,contributing%2010.7%25%20of%20new%20cases

[10] G. Ke, “Features,” Features - LightGBM 3.3.3.99 documentation. [Online]. Available: https://lightgbm.readthedocs.io/en/latest/Features.html. [Accessed: 17-Jan-2023].

RMIT UNIVERSITY