IEEE International Black Sea Conference on Communications and Networking

Android Malware Detection Using Incremental Learning Approach

By

Shawon Kumar Saha
Dept. of CSE, BUET

Md. Shohrab Hossain
Dept. of CSE, BUET

Rafidul Islam Sarker
Dept. of CSE, BUET

Husnu S. Narman
Dept. of CSEE
Marshall University, USA

Introduction

- Android is a linux based mobile operating system
- More than 3 billion global users
- Around 70% of android smartphone users

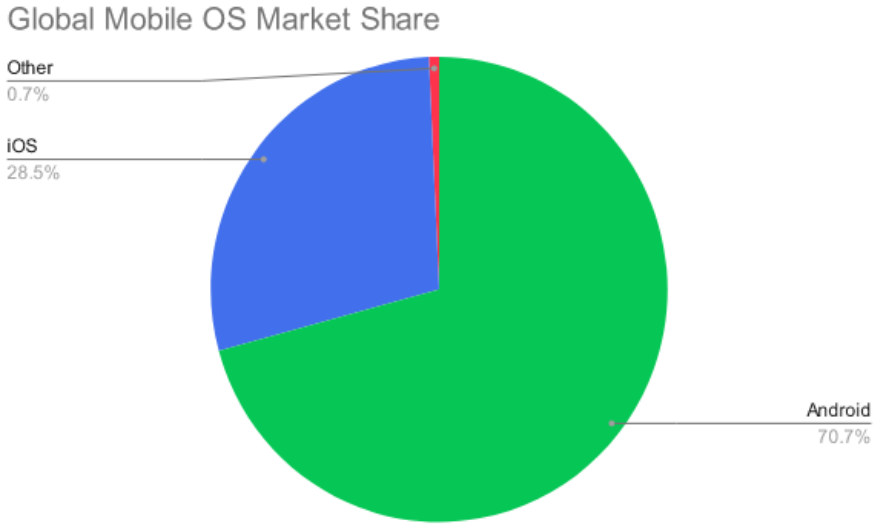

Figure 1: Global mobile market

Introduction

- Android applications are developing rapidly across the mobile ecosystem
- Malware is portion of code that is written with the intention of harming people
- Android malware is also emerging in an endless stream.Over 10,000 new malware samples per day

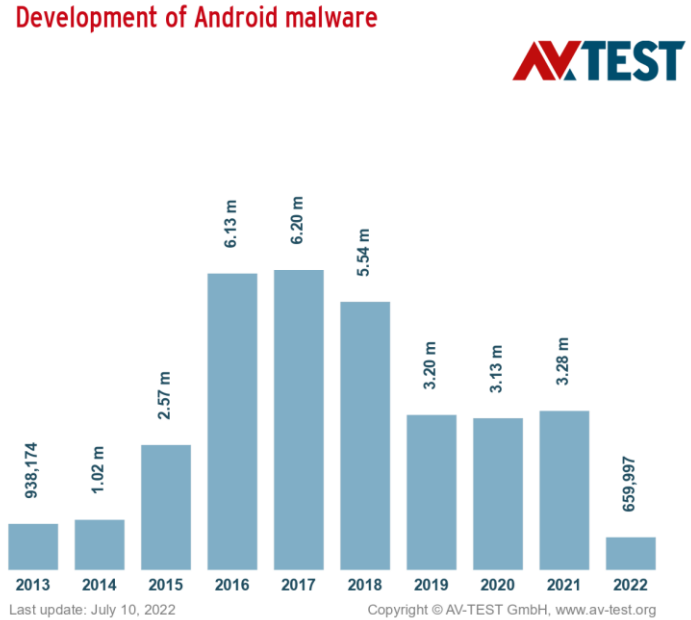

Figure 2: Android malware per year

Android Malware

- Activity
  - Steal personal info such as contact, bank account and other sensitive info
  - Use for ddos botnet
  - Ransom
  - Crypto mining

Motivation

- Malware collect personal data without one's consent
- Over 90 million mobile banking users in bangladesh
- Early detection helps to stop unauthorized access of personal data

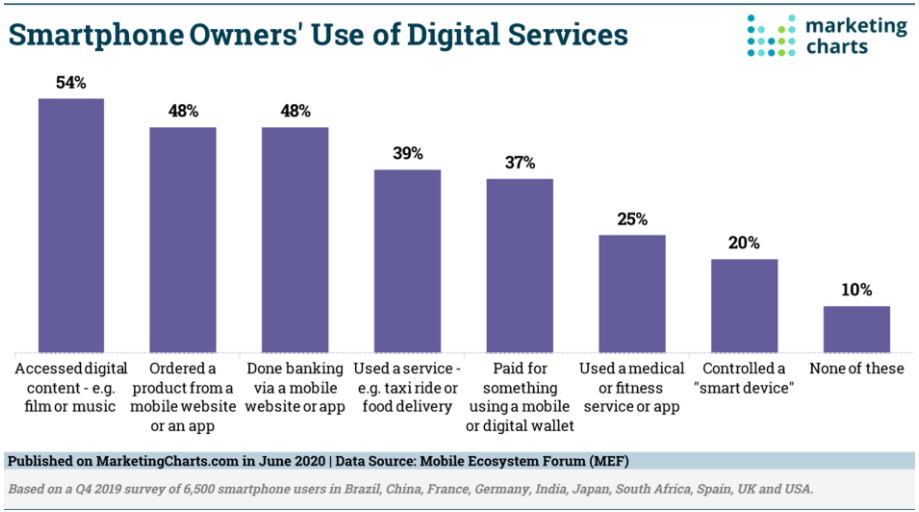

Figure 3: Activities of smartphone

Malware Detection

- Static
  - Review apk files to find patterns
- Dynamic
  - Monitoring runtime behavior of applications
- Hybrid
  - Monitoring both Static and Dynamic features

Literature Review

RanDroid:Android Malware Detection Using Random Machine Learning Classifiers[1] By J. D. Koli [ICSESP-2018]

- Requested permissions, Vulnerable API calls, Presence of key information
- Classification Algorithm (SVM, DT, RF, NB)
- Advantages
  - Perform better than their reviewed pape having better accuracy and F-measure.
- Disadvantages
  - Dataset is small having 120 benign, 175 malicious application and missed many features

Literature Review

ReDroidDet: Android Malware Detection Based on Recurrent Neural Network[2] By Almahmouda et al. [Procedia Computer Science, 2022]

- Considered Features
  - Permissions, API calls, system events, permission rate
- Considered Classification Algorithm
  - SVM, KNN, NB, RF, DT, RNN
- Advantages
  - Balance dataset and better than review papers
- Disadvantages
  - They consider a few no. of top level features

Literature Review

DATDroid: Dynamic Analysis Technique in Android Malware Detection[3] by Thangavelooa et al. [IJASEIT, 2020]

- System call, CPU & memory usage, Network packets
- Gain Ratio Attribute Evaluator for feature selection
- Random Forest classifier
- Advantages
  - Use different combination of the extracted features to obtain higher accuracy
- Disadvantages
  - Used a small dataset considering a few number of features

Literature Review

A TAN based hybrid model for android malware detection [4] by Surendran et al. [JISA, 2020]

- Used hybrid malware detection mechanism
- Use Static (API Calls, Permission) & Dynamic (System call) features
- Ridge regularized logistic regression classifier and Tree Augmented naive Bayes
- Advantage
  - balanced data, better accuracy, modern malware behaviour
- Disadvantages
  - Malwares escape
  - Less no. of feature and no powerful model

Overall Situation

- Imbalanced Class samples
- Limited feature selection
- Lack of combinations of features
- Lack of implementations of online learning approaches

Solution

- Proper balancing of class data
  - Using SMOTE
- Using more static features
- Selection of different categories of features
- Online learning based classification

Dataset

App Type	Number of Sample	Number of Features
Adware	119	2109
Ramsomware	101	1093
Scareware	111	1955
Smsmalware	107	1577
Benign	599	4504

Table 1: Sample and Feature number of different Apps

App Type	Number of Sample
Adware	599
Ransomware	160
Scareware	159
Smsmalware	139
Benign	141

Table 2: Data sample’s class distribution

Proposed Model

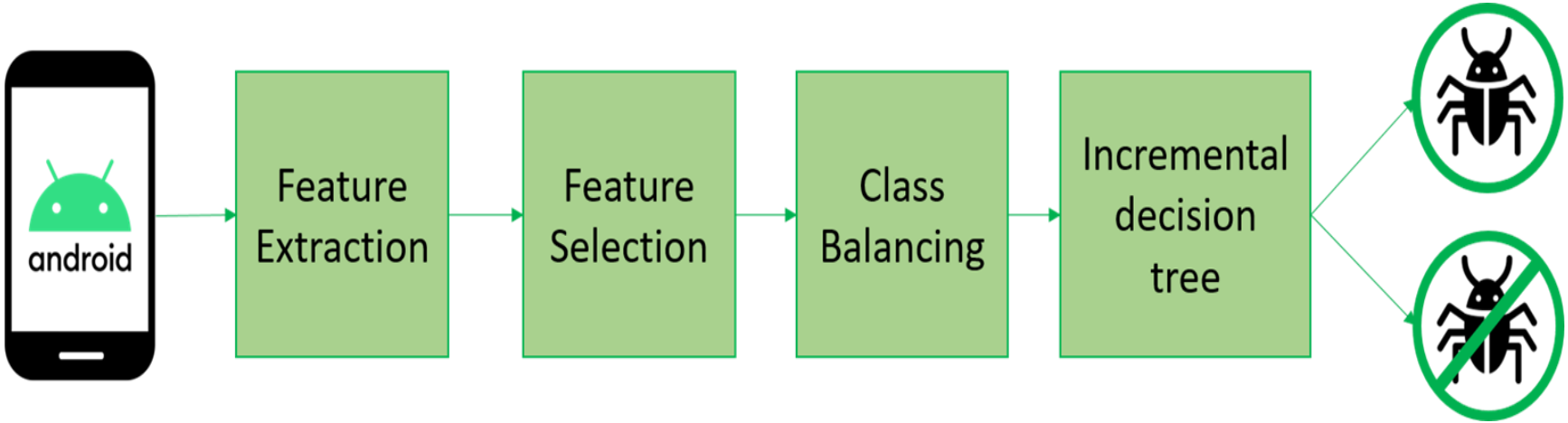

Figure 4: Proposed workflow model

Incremental Decision Tree

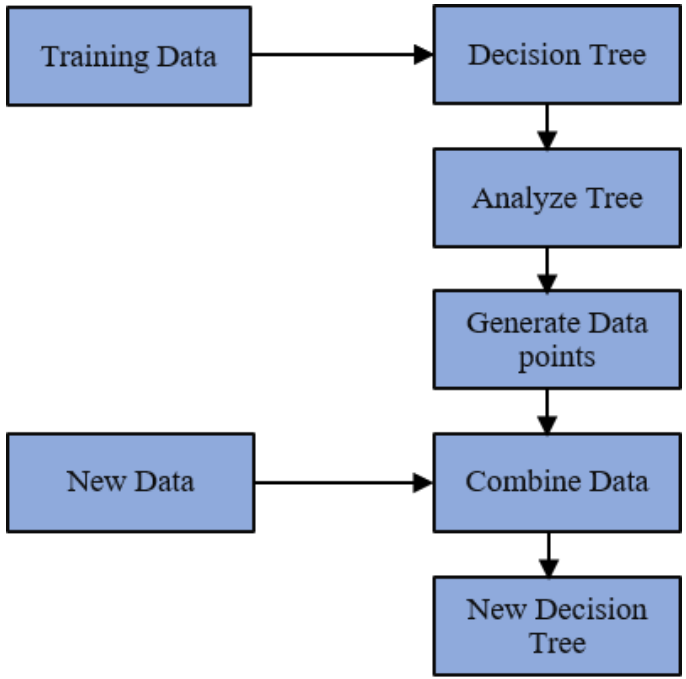

Figure 5: Incremental Decision Tree

Difference

- Balance Data set
- Consider online learning based method
- Want to add more types of features

Implementation

- Used andropytool for extracting APK's features
- AndroPyTool
  - Extract Static and Dynamic feature from Android APK
  - Combine different android app analysis tools like
    - DroidBox
    - AndroGuard
- Generate files of features in JSON and CSV formats

Implementation

- Calculate information gain
- Use entropy of classes to calculate information gain
- Save all features' information gain into List
- Use the top features for future analysis

Implementation

Balance Dataset

- using SMOTE, choose random sample and its one nearest neighbour
- Choose a synthetic data point between them
- Make synthetic data points of minority classes

Result Analysis

- We Apply LGBM model on our dataset
- 92.92% accuracy on our dataset
- 96.75% precision on our dataset
- 90.15% recall on our dataset
- 93.33% f1-score on our dataset

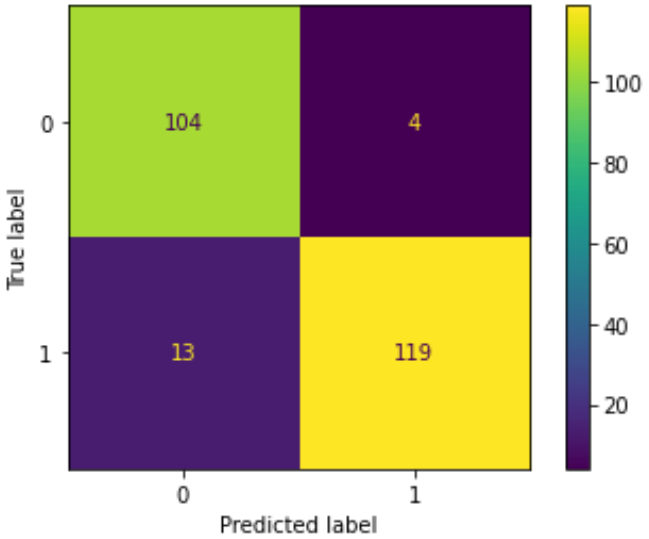

Figure 6: Confusion matrix of LGBM

Result Analysis

- We Apply XGBoost model on our dataset
- 92.92% accuracy on our dataset
- 94.57% precision on our dataset
- 92.42% recall on our dataset
- 93.49% f1-score on our dataset

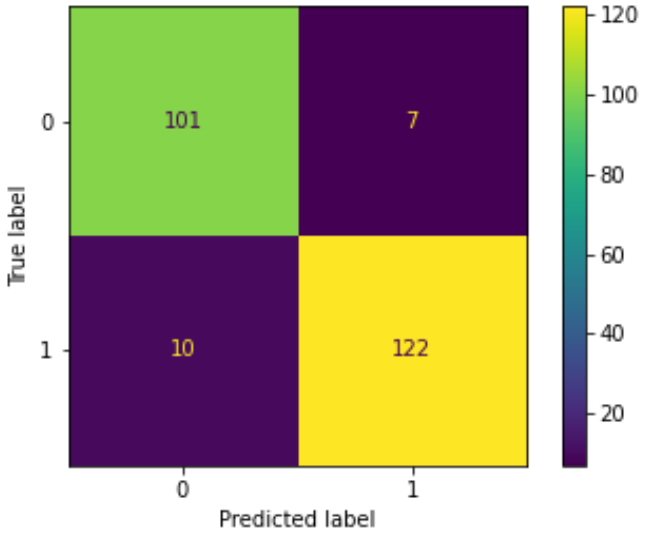

Figure 7: Confusion matrix of XGBoost

Result Analysis

- We Apply Decision Tree model on our dataset
- 89.58% accuracy on our dataset
- 91.47% precision on our dataset
- 89.39% recall on our dataset
- 90.42% f1-score on our dataset

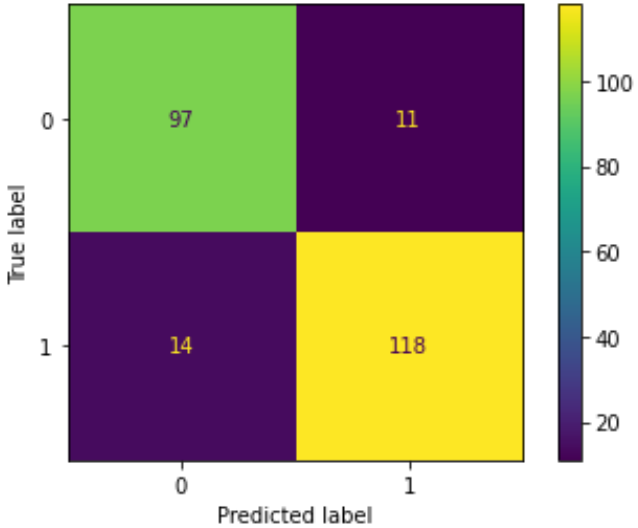

Figure 8: Confusion matrix of Decision Tree

Result Analysis

- We Apply our proposed Incremental Decision Tree model on our dataset
- 93.33% accuracy on our dataset
- 91.27% precision on our dataset
- 95.83% recall on our dataset
- 93.50% f1-score on our dataset

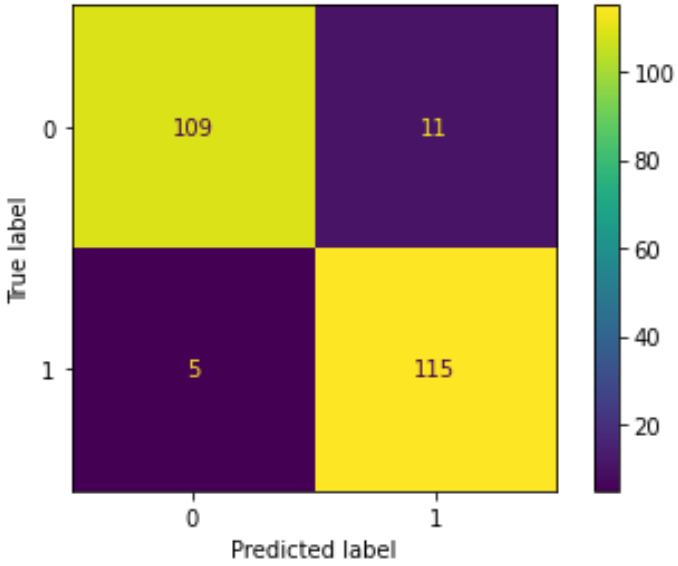

Figure 9: Confusion matrix of Incremental Decision Tree

Result Analysis

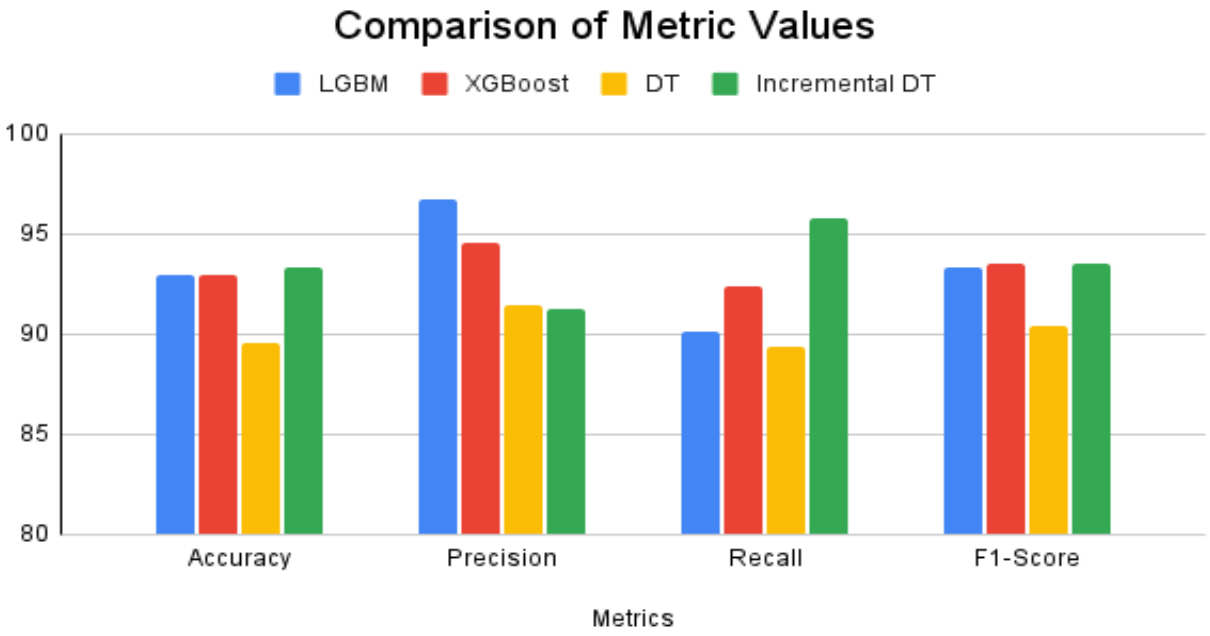

Figure 10: Comparison of different metric values

Result Analysis

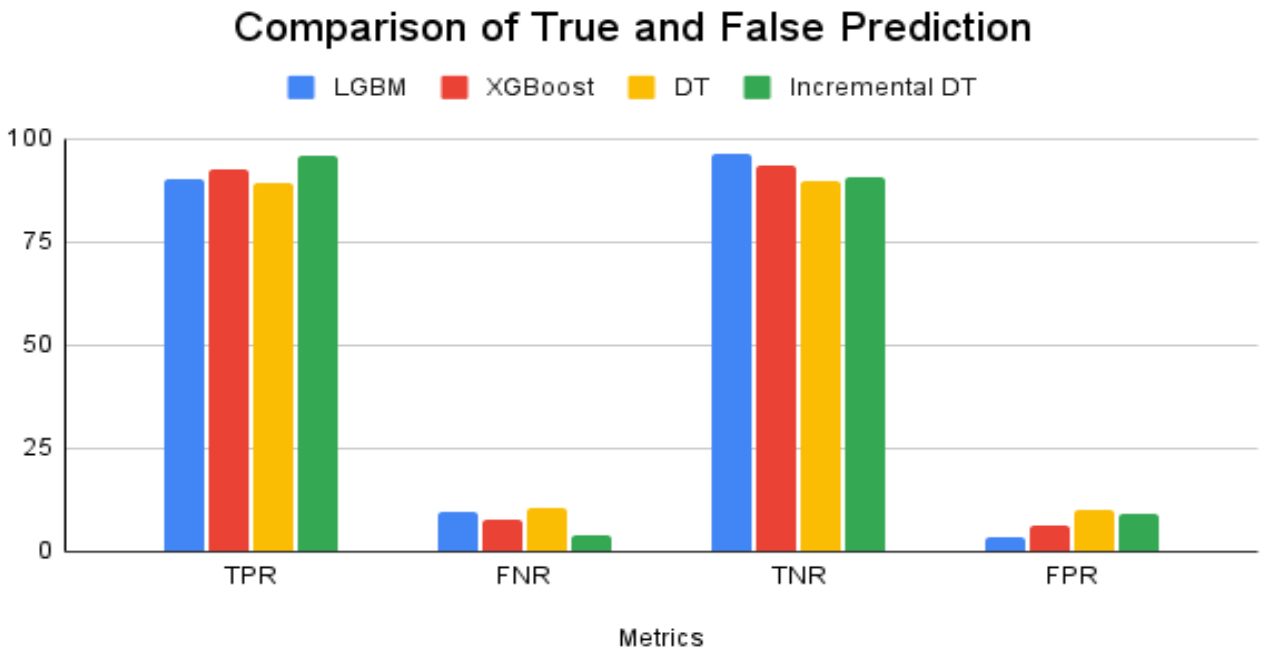

Figure 11: Comparison of different metric values

Conclusion

- Our incremental model achieve higher accuracy
- Incremental model achieve higher recall value
- Incremental model achieve higher f1-score
- Perform better in real time data scenarios

Future Works

- Complete Implementation of other incremental models
- Extract dynamic features from applications
- Consider larger dataset

Reference

[1] Koli, J. D. "RanDroid: Android malware detection using random machine learning classifiers." *2018 Technologies for Smart-City Energy Security and Power (ICSESP)*. IEEE, 2018.

[2] Almahmoud, Mothanna, Dalia Alzu'bi, and Qussai Yaseen. "ReDroidDet: android malware detection based on recurrent neural network." Procedia Computer Science 184 (2021): 841-846.

Reference

[3] Thangavelooa, Rajan, et al. "Datdroid: Dynamic analysis technique in android malware detection." *International Journal on Advanced Science, Engineering and Information Technology* 10.2 (2020): 536-541.

[4] Surendran, Roopak, Tony Thomas, and Sabu Emmanuel. "A TAN based hybrid model for android malware detection." *Journal of Information Security and Applications* 54 (2020): 102483.

Thank You