

APRIL 28, 2023

ML SOFTWARE VULNERABILITIES

PREPARED BY: GROUP [TEAM 15]

HARSHITA JAIN

NAMRATA GAUR

SHREYA REDDY BOLLA

TAPASWI REDDY BUSIREDDY

VINITA MALOO

SPONSOR: FOTEINI ARGIROPOULOS

TABLE OF CONTENTS

Project Description.....	2
User Manual.....	2
Installation Guide.....	21
Issues.....	21
Outlook.....	21
Video Presentation.....	22

PROJECT DESCRIPTION

Static code analyser tools are used to detect the code smells in a source code. However, they do not assess the vulnerabilities present in the source code. Vinci Tool is a standalone machine learning application that can be used to predict vulnerabilities in a source code. The tool is a machine learning model that is trained based on a static code analyser report(either in XML or CSV format) provided to it. After successfully training the model, the tool makes its predictions of the vulnerabilities and assesses whether an entity in the static code analyser tool is a vulnerability or not. In addition, the tool can also be used to assess the risk(threat level) of software vulnerabilities. The tool helps in finding and avoiding vulnerabilities in software which can help prevent cyber attacks.

USER MANUAL

The several features that were implemented are:

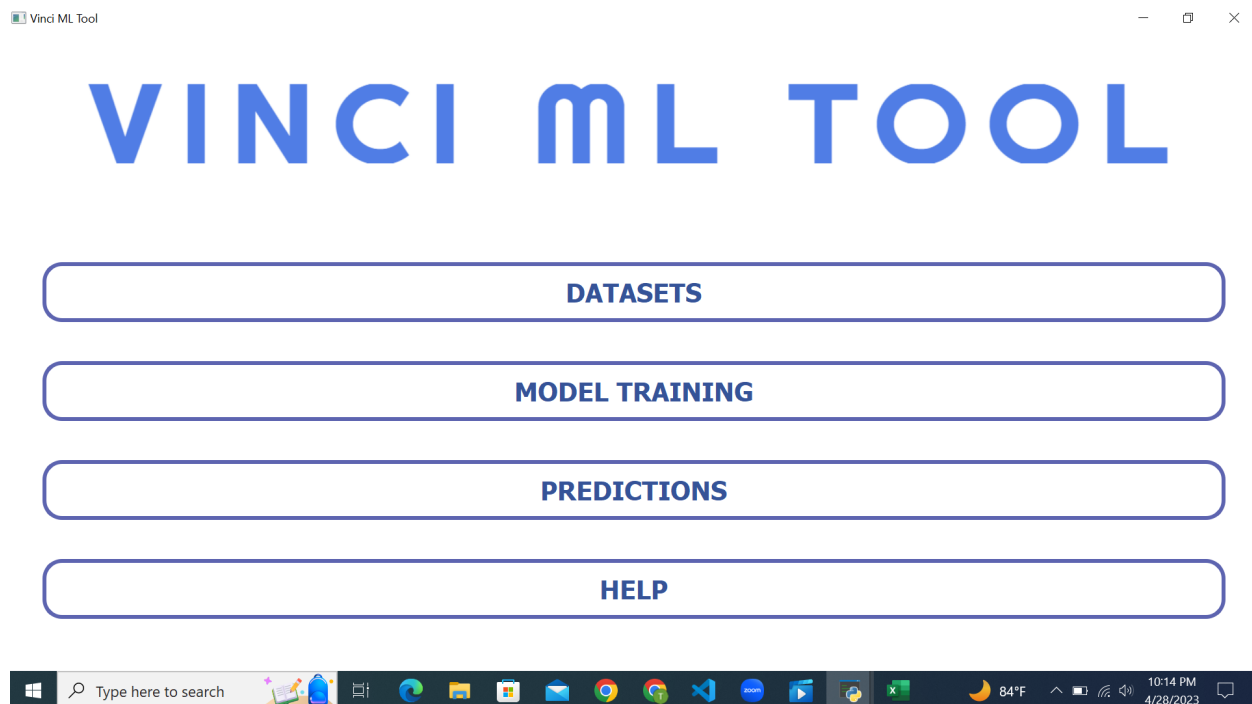
1. Static code analyser tool support:

a. Description:

As users and companies prefer to use varied programming languages, it is essential to provide support to more than one static code analyser tool in the Vinci tool. Therefore, we provided initial support to reports from various static code analysers, such as PMD(for Java), PHP_Codesniffer(for PHP), Pylint(for Python), ESLint, and JSHint(for Javascript). This will help users from several backgrounds to prefer the tool.

b. Feature user guide:

- i. Run the tool.
- ii. On the landing page, choose the 'Datasets' button.



iii. In the 'Datasets' window, select the 'Preset' option on the right side.

Dataset - Columns and Types

Root: Add Column

Column: Remove Column

Data: Training: Transformation: Type: TP Label:

	Column	Use for training?	Data	Transformation	Type
1	raw	Yes	Categorical	One-hot Encoding	Output (escalated)
2	evidence	Yes	Categorical	One-hot Encoding	Input
3	line	Yes	Categorical	One-hot Encoding	Input
4	character	Yes	Categorical	One-hot Encoding	Input
5	scope	Yes	Categorical	One-hot Encoding	Input
6	reason	Yes	Categorical	One-hot Encoding	Input

Loading JSON columns, please wait...

Dataset - Information

2 Samples

6 Features

Preset:

Dataset - Stratified Sampling

10 90

% Train % Test

Train Test

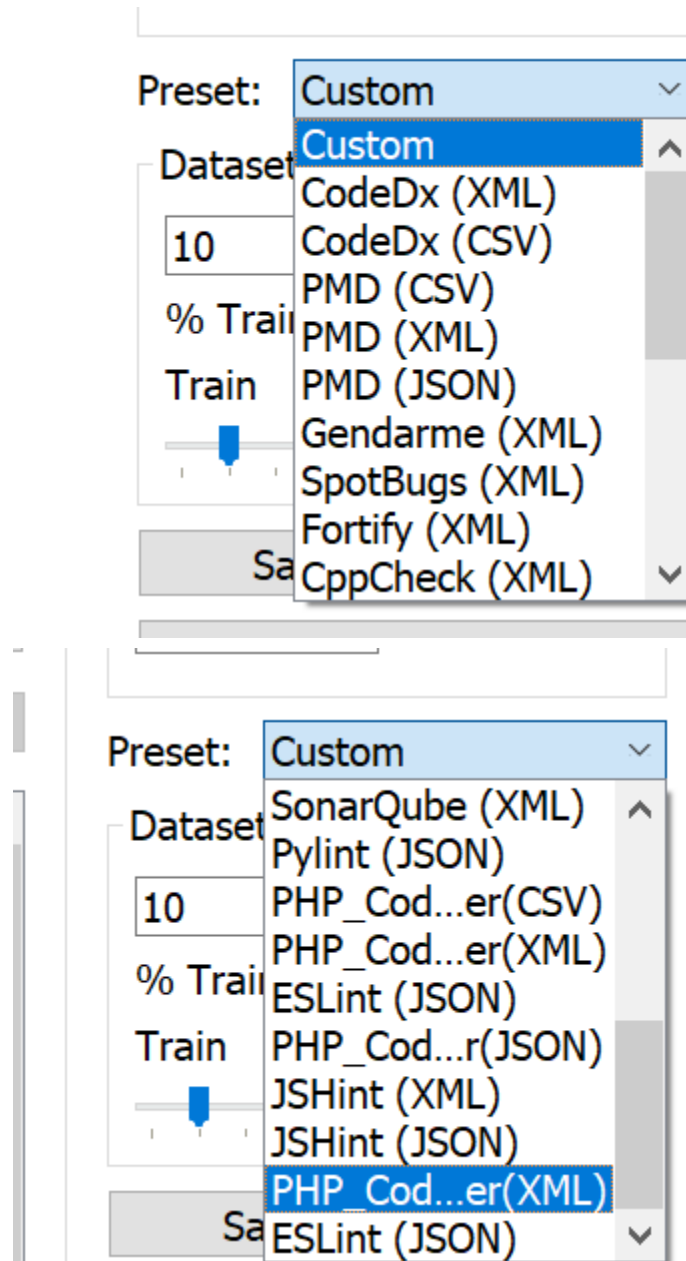
Save New Dataset

Save Schema

Load Schema

Help

- iv. The 'Preset' dropdown shows the various static code analyser tools that are supported by the Vinci tool.



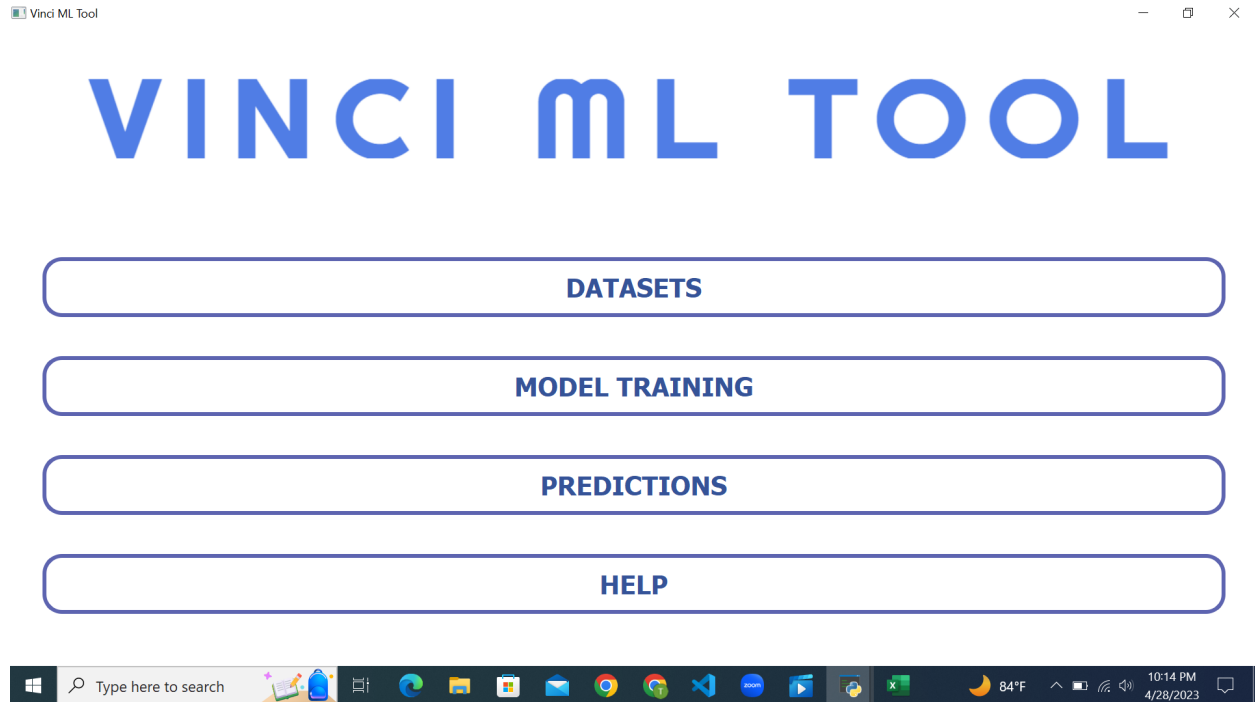
- v. In the 'Datasets' window, click on the 'Load Dataset' button to load a dataset of your choice.
- c. **Sponsor requirement:** Yes

2. Different file formats support:**a. Description:**

Initially the tool only supported the XML, and CSV file formats for the report. So, we have added JSON file format as well to load and label the data in the tool. Now, the tool supports XML, and CSV to the full extent with all the datasets and the reports but with JSON we could only load and label the data.

b. Feature user guide:

- i. Run the tool.
- ii. On the landing page, choose the 'Datasets' button.



iii. In the 'Datasets' window, select the 'Load Dataset' option on the left side.

Datasets | Vinci ML Tool

Load Dataset Path: D:/SER-517-Team-15-Spring-2023/ML4CyberVinciPython-main/input/JSHint_JSON.json

Dataset - Columns and Types

Root: Add Column

Column: Remove Column

Data: Training: Transformation: Type: TP Label:

	Column	Use for training?	Data	Transformation	Type
1	raw	Yes	Categorical	One-hot Encoding	Output (escalated)
2	evidence	Yes	Categorical	One-hot Encoding	Input
3	line	Yes	Categorical	One-hot Encoding	Input
4	character	Yes	Categorical	One-hot Encoding	Input
5	scope	Yes	Categorical	One-hot Encoding	Input
6	reason	Yes	Categorical	One-hot Encoding	Input

Dataset - Information

2 Samples

6 Features

Preset: JSHint (JSON)

Dataset - Stratified Sampling

10 90

% Train % Test

Train Test

Save New Dataset

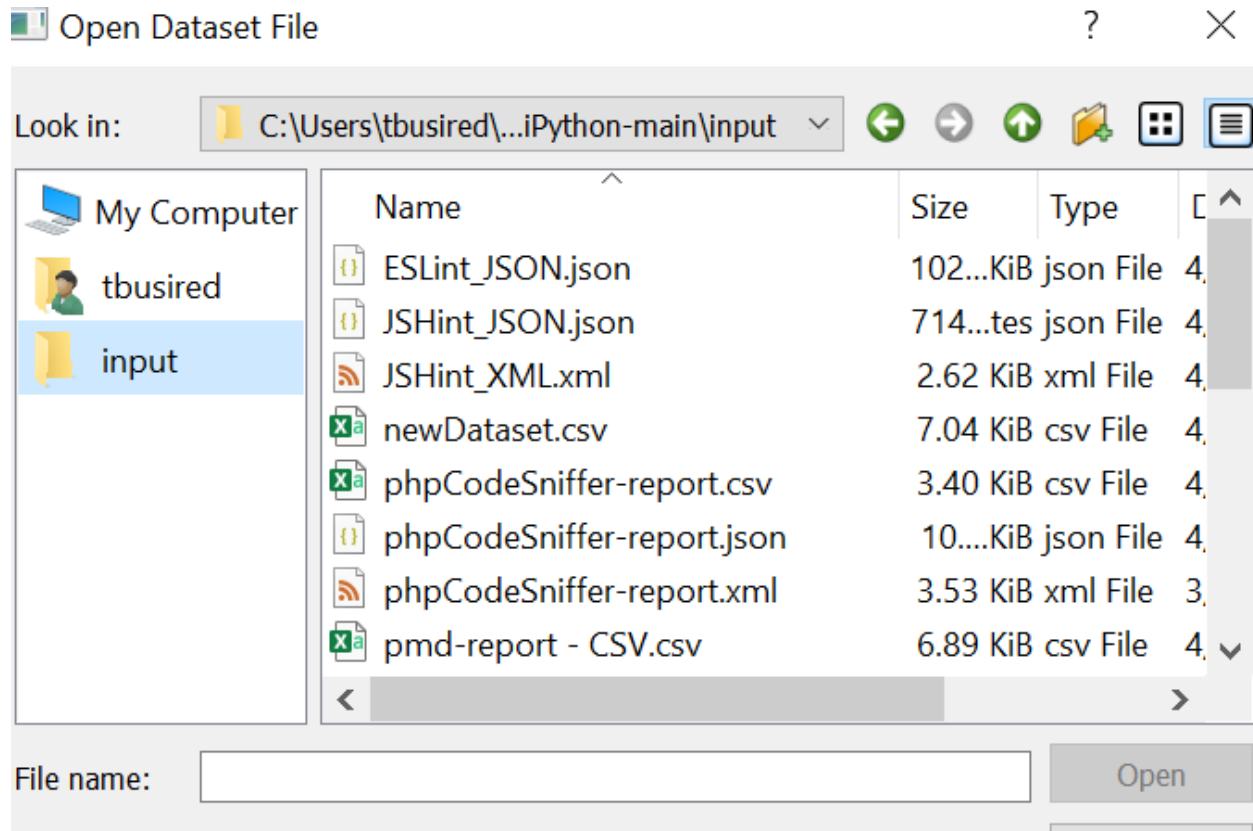
Save Schema

Load Schema

Help

Loading JSON columns, please wait...

- iv. The 'Load Dataset' options let you select the file from the file formats that are supported by the tool.



- c. **Sponsor requirement:** Yes

3. Dynamic risk analysis:

a. Description:

The tool initially had the option to assess the risk of each entity from the input report separately. To facilitate dynamic risk analysis, a logistic regression model was used. The user can label a few of the entities and save the results which will be considered as the training set for the model. A JSON-formatted schema is generated based on the training data which is then fed to the model. Then, a model is generated based on the schema and training data. Finally, the saved dataset, the schema, and the model are used to make predictions and the risk level results are saved in a CSV file.

b. Feature user guide:

- i. Run the tool.
- ii. On the landing page, choose the 'Datasets' button.

VINCI ML TOOL

DATASETS

MODEL TRAINING

PREDICTIONS

HELP



- iii. In the 'Datasets' window, click on the 'Load Datasets' button and load a CodeDx CSV dataset.

Datasets | Vinci ML Tool

Load Dataset Path: D:\SER-517-Team-15-Spring-2023\ML4CyberVinciPython-main\input\JSHint_JSON.json

LABELER

Dataset - Information

2 Samples

6 Features

Preset: JSHint (JSON)

Dataset - Stratified Sampling

10 90

% Train % Test

Train Test

Save New Dataset

Save Schema

Load Schema

Help

Dataset - Columns and Types

Root: Add Column

Column: raw Remove Column

Data: Categorical Training: Yes Transformation: One-hot En Type: Input TP Label:

	Column	Use for training?	Data	Transformation	Type
1	raw	Yes	Categorical	One-hot Encoding	Output (escalated)
2	evidence	Yes	Categorical	One-hot Encoding	Input
3	line	Yes	Categorical	One-hot Encoding	Input
4	character	Yes	Categorical	One-hot Encoding	Input
5	scope	Yes	Categorical	One-hot Encoding	Input
6	reason	Yes	Categorical	One-hot Encoding	Input

Loading JSON columns, please wait...

- iv. Click on the 'Labeler' button.
- v. In the 'Labeler' window, click on the 'Risk' button.

Datasets - Labeler | Vinci ML Tool

Dataset Sampling

Load Dataset Path: C:/Users/tbusired/Downloads/new_dataset.csv

620 # Rows Size 62 Size (%) 10.00 Type: Random Save Sample

Custom Seed 0 Invert Sample Ref:

Dataset Labeling

Load Dataset Path: C:/Users/tbusired/Downloads/new_dataset.csv

Column Label: output True Label: True Positive False Label: False Positive **RISK**

	Key	Value
1	@status	escalated
2	@severity	info
3	cwe->@id	
4	results->result->description->#text	Unnecessary semicolon.

Current Example: True

Sample: 1

Progress: 1/620 (0.16 %)

< >

Save Dataset Help

Page: 1 / 25

- vi. In the 'Risk' window, change the values of the scores as necessary and click on 'Save Scores' to save the threat level for that entity.

Risk Assessment | Vinci ML Tool

Current Score

Base Finding Subscore: 43.20 Environmental Subscore: 0.38 Threat Level:

Attack Surface Subscore: 0.73 **Final CWSS Score: 12.15 Low**

? Path Column: @status Help Cancel/Close

Sample Info Base Finding Attack Surface Environmental CWSS Score

Group Labeling

File: Disabled ? Values: @status ?

Folder: Disabled ? Add Remove

	Key	Value
1	@status	escalated
2	@severity	info
3	cwe->@id	

Page: 1 / 25 Sample: 1 / 620

Save Score Save Results Progress: 0.0 < > Save Schema Model Predict



Risk Assessment | Vinci ML Tool

Current Score

Base Finding Subscore: 48.00 Environmental Subscore: 1.00 Threat Level:

Attack Surface Subscore: 0.80 Final CWSS Score: 38.16 Low

? Path Column: @status Help Cancel/Close

Sample Info Base Finding Attack Surface Environmental CWSS Score

Group Labeling

File: Disabled ? Values: @status ?

Folder: Disabled ? Add Remove

Key	Value
1 @status	escalated
2 @severity	info
3 cwe->@id	

Page: 1 / 25 Sample: 3 / 620

Save Score Save Results Progress: 3/620 (0.48 %) < > Save Schema Model Predict

- vii. After sufficient entries have been labeled, click on the 'Save Results' button and save the CSV file.

Save Risk Results File

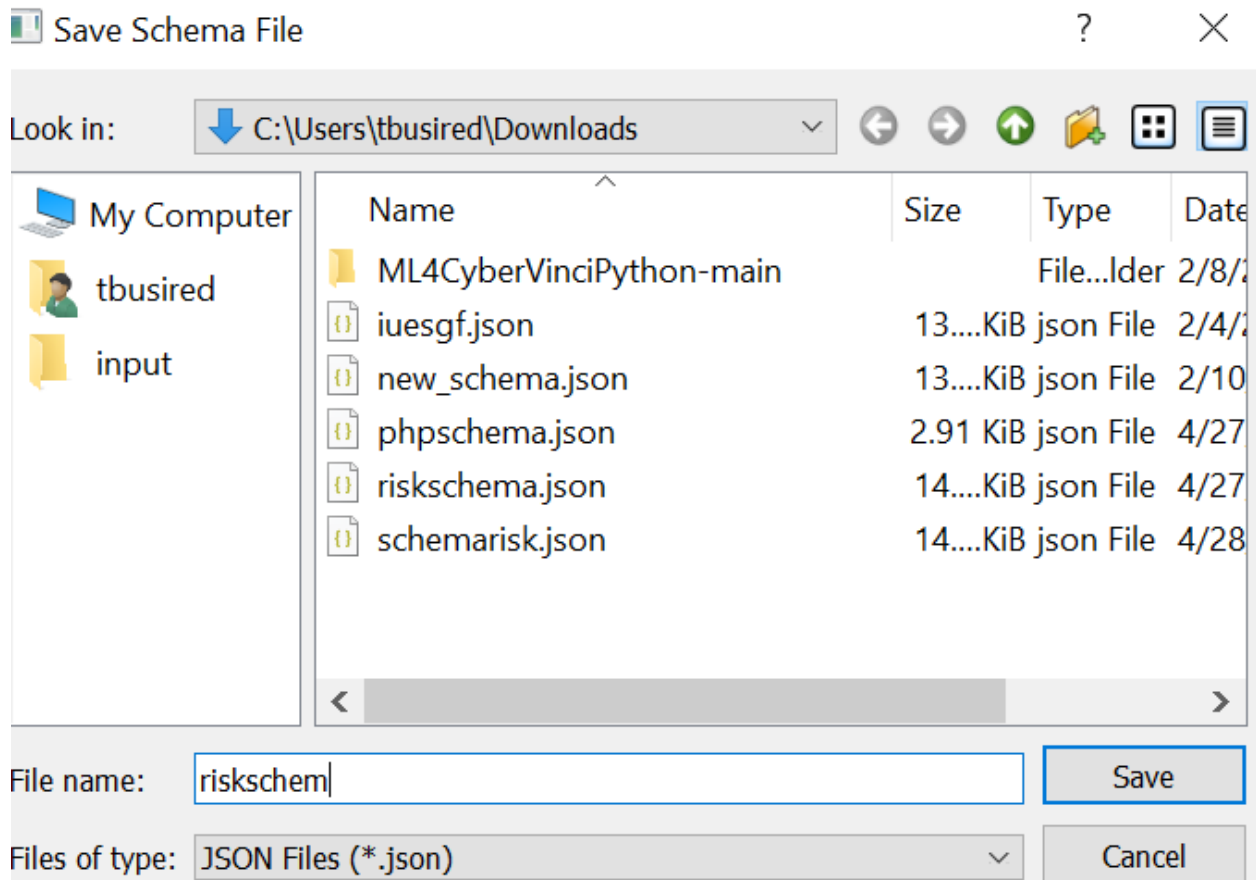
Look in: C:\Users\tbusired\Downloads

Name	Size	Type
ML4CyberVinciPython-main		File...lder 2
convertedCSV.csv	5.38 MiB	csv File 2
new_dataset.csv	78....KiB	csv File 4
new_sample.csv	7.63 KiB	csv File 2
phpCodeSniffer-report.csv	3.40 KiB	csv File 4
phpresults.csv	3.91 KiB	csv File 4
phptest.csv	3.51 KiB	csv File 4
pmd-report.csv	6.79 KiB	csv File 4

File name: resultrisk Save

Files of type: CSV Files (*.csv) Cancel

viii. Click on 'Save Schema' and save the JSON formatted schema file.



- ix. Now, click on the 'Model' button.
- x. In the 'Model' window, upload the previously saved schema file and change the epochs, learning rate, and threshold as necessary.

Models | Vinci ML Tool

Load Schema Path:

Models - Training Parameters

Epochs:

Learning Rate:

Metrics

☒ Accuracy ☐ Precision ☐ Recall

Dataset - Information

Features

Train Samples

Test Samples

Models - Train and Save

Train Now Progress: | **Save Model** **Help**

24%

Evaluate

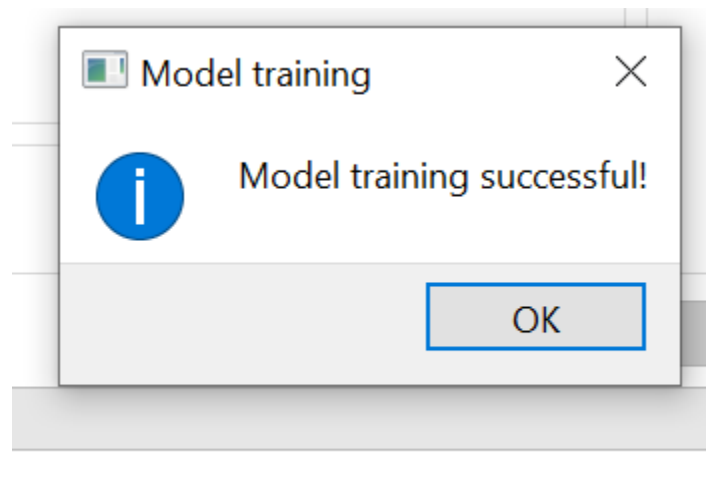
Dataset Sample:

Metrics

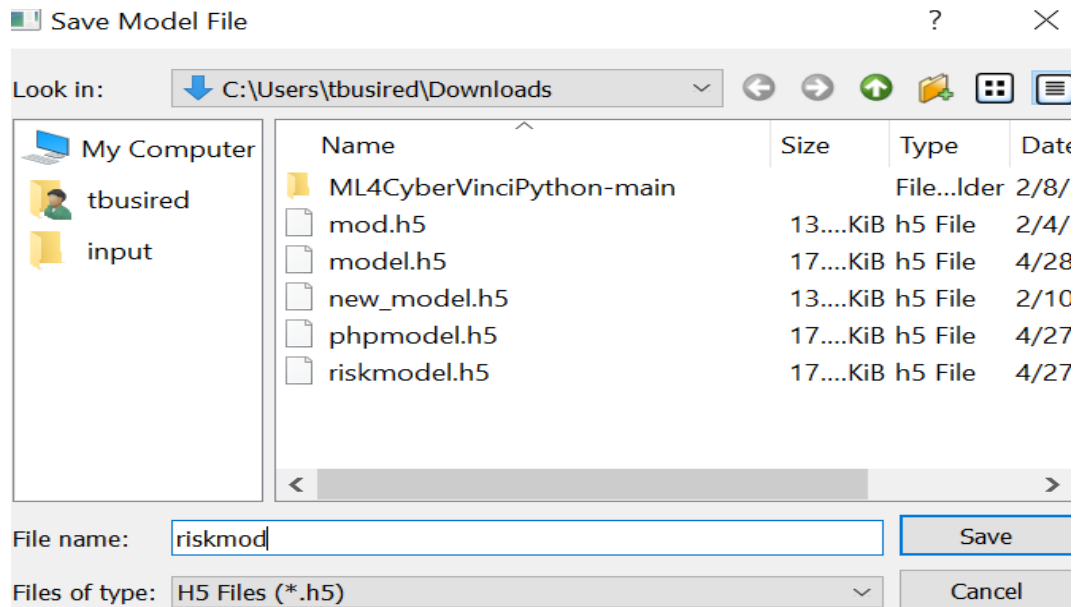
Threshold:

	True Positive	True Negative	False Positive	False Negative
Loss	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
	Accuracy	Precision	Recall	F1 Score

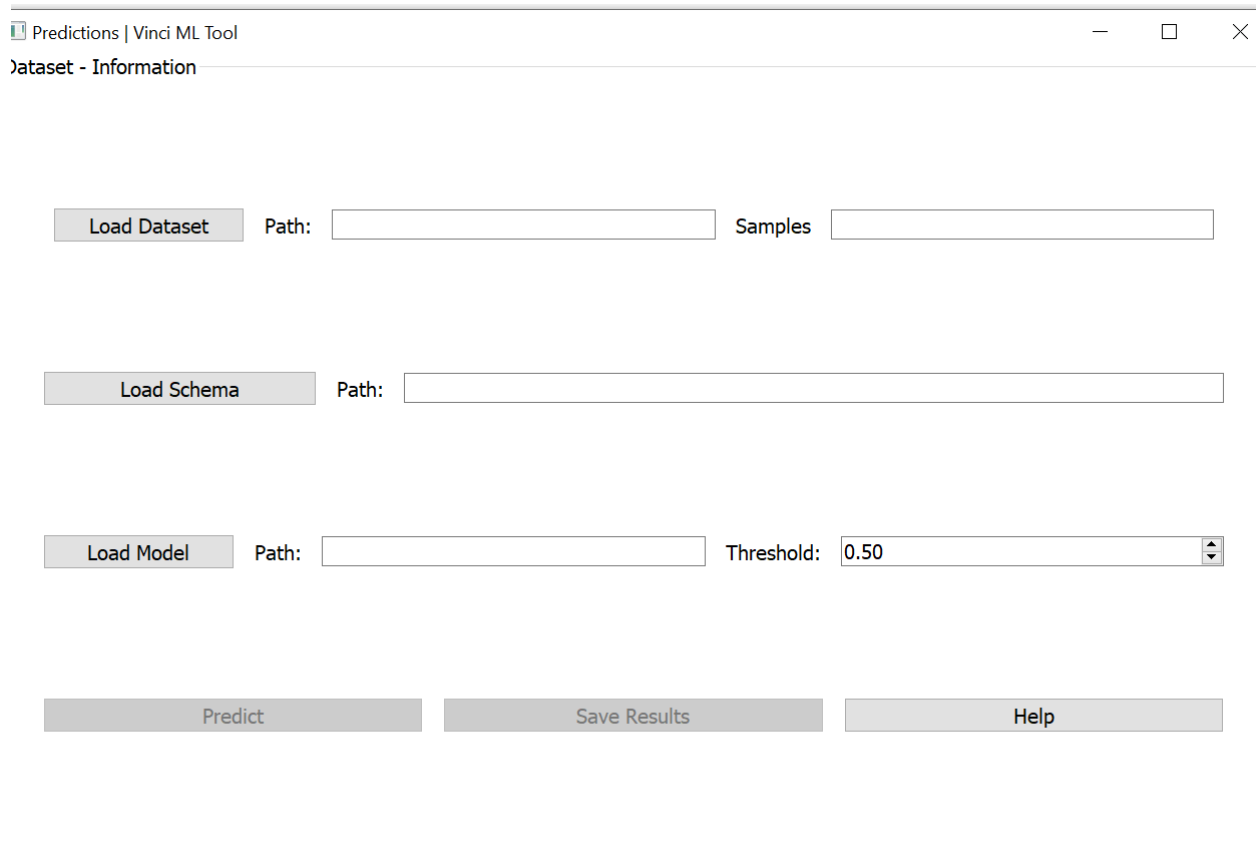
- xi. Click on 'Train Now' and wait for the model to be trained.



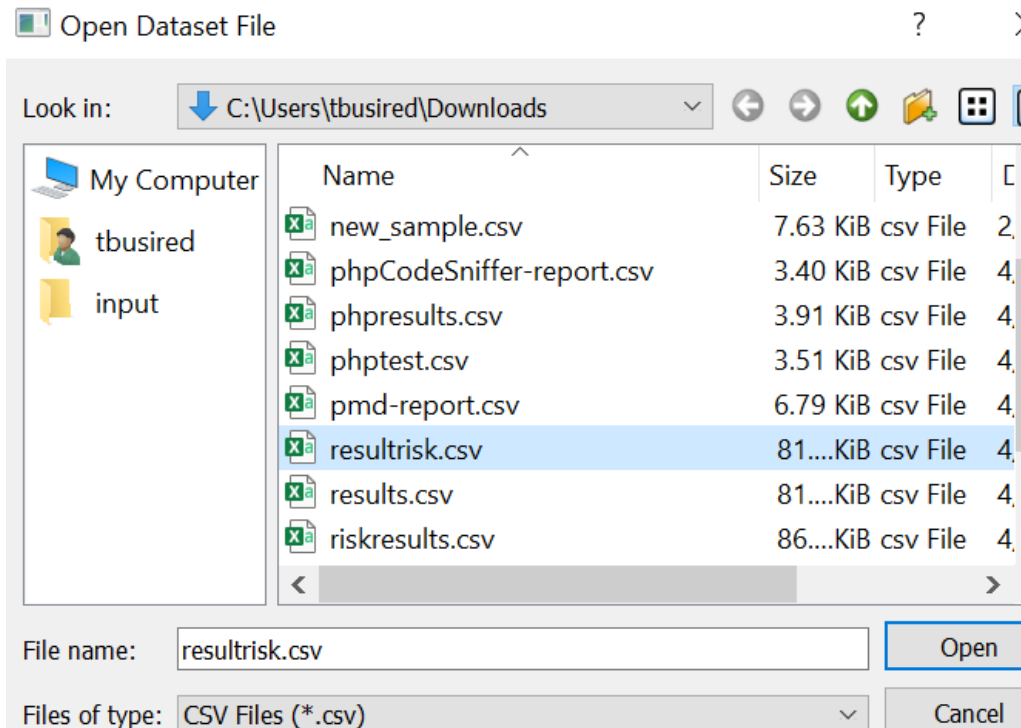
xii. When the training is finished, click on 'Save Model' and save the model.



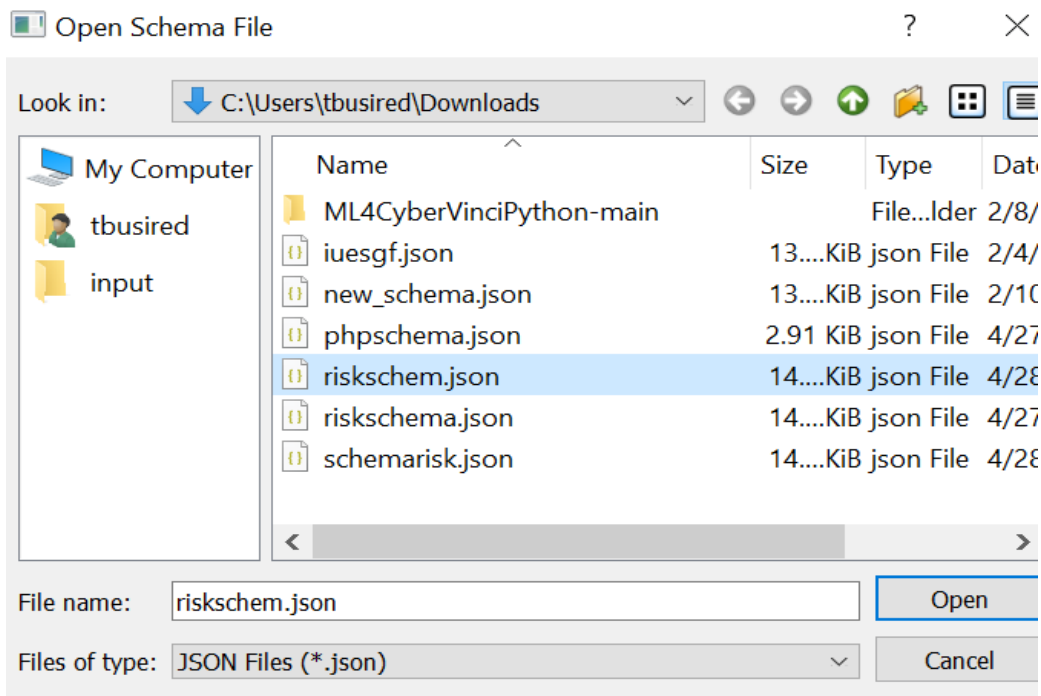
xiii. Now click on the 'Predict' button in the 'Risk' window.

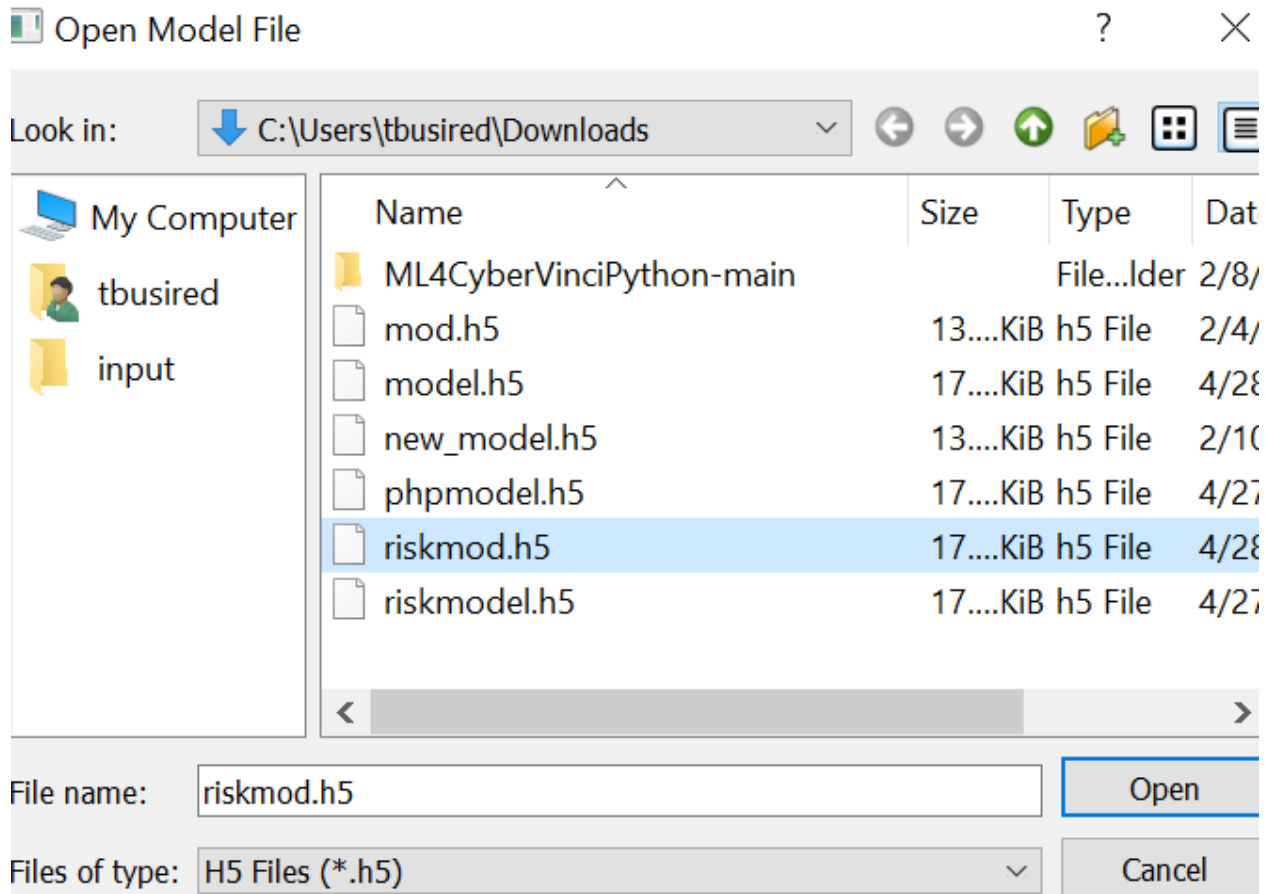


- xiv. In the 'Prediction' window, click on 'Load Dataset' and load the saved dataset from step vii.

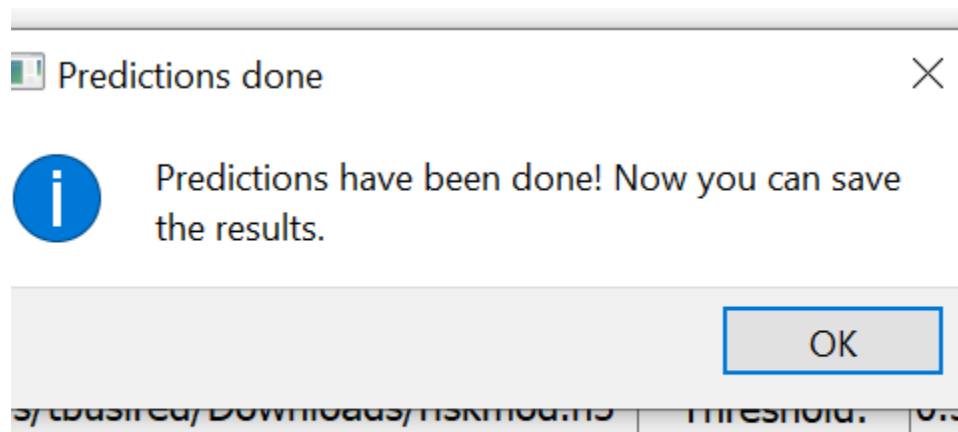


- xv. Click on 'Load Schema' and load the schema from step viii and click on 'Load Model' and load the model from step xii.

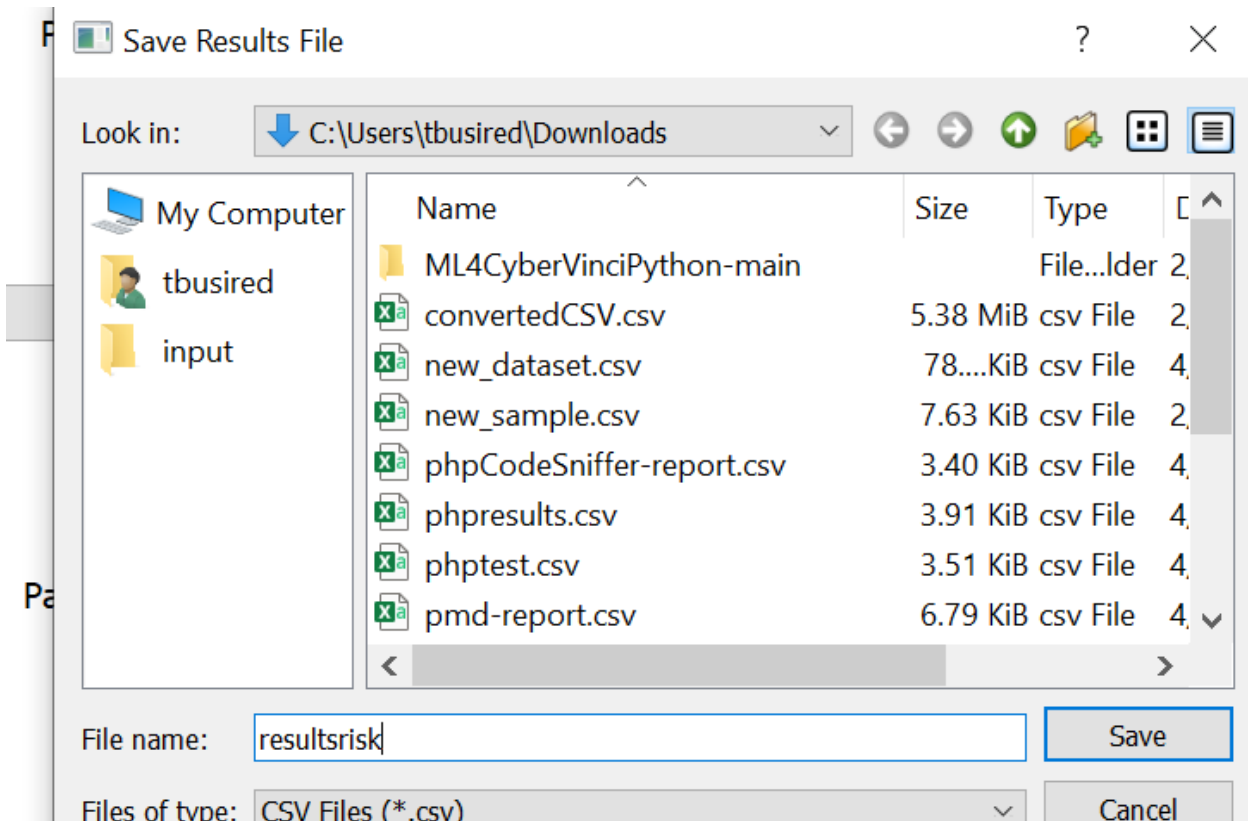




xvi. Click on 'Predict' and wait for predictions to be done.



xvii. Click on 'Save Results' to save the results.



xviii. A new column with the predictions will be visible in the saved CSV file.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
453	451	451	451	escalated	high	77				rule-5e8e!Command Injection	high	low							
454	452	452	452	escalated	high	77				rule-5e8e!Command Injection	high	low							
455	453	453	453	escalated	high	77				rule-5e8e!Command Injection	high	low							
456	454	454	454	escalated	high	77				rule-5e8e!Command Injection	high	low							
457	455	455	455	escalated	high	77				rule-5e8e!Command Injection	high	low							
458	456	456	456	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
459	457	457	457	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
460	458	458	458	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
461	459	459	459	escalated	high	77				rule-5e8e!Command Injection	high	low							
462	460	460	460	escalated	high	77				rule-5e8e!Command Injection	high	low							
463	461	461	461	escalated	high	77				rule-5e8e!Command Injection	high	low							
464	462	462	462	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
465	463	463	463	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
466	464	464	464	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
467	465	465	465	escalated	high	77				rule-5e8e!Command Injection	high	low							
468	466	466	466	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
469	467	467	467	escalated	high	77	Generic O	security	ESLint	rule-5e8e!Command Injection	high	high							
470	468	468	468	escalated	high	77				rule-5e8e!Command Injection	high	low							
471	469	469	469	escalated	high	77				rule-5e8e!Command Injection	high	low							
472	470	470	470	escalated	high	77	Function C	security	ESLint	rule-5e8e!Command Injection	high	high							

c. **Sponsor requirement:** Yes

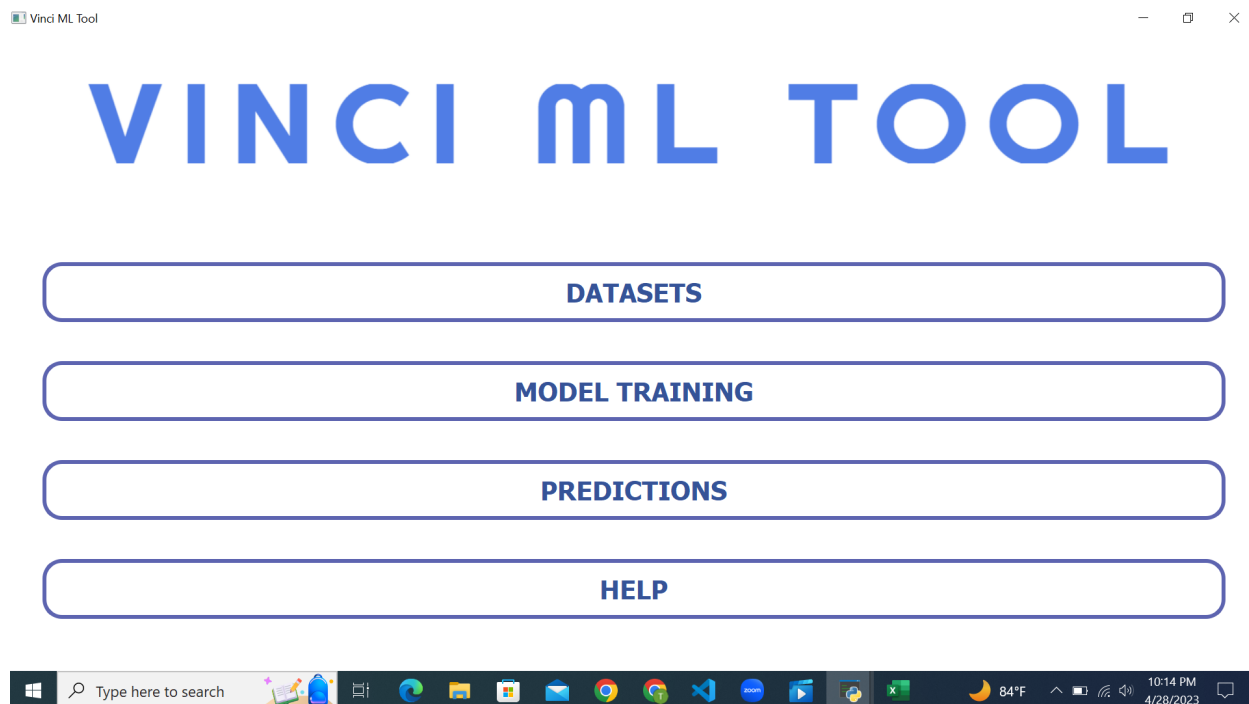
4. Dynamic window resizing:

a. Description:

Initially the windows were very fixed and the components in the window were very congested and wouldn't resize while maximizing. So, we added a dynamic resizing to the window which expands whenever the user clicks on the maximized window. In, this way the components look larger for the windows when maximize and fit right according to the laptop display screen.

b. Feature user guide:

- i. Run the tool.
- ii. Click on maximize to view the window size



- iii. Click on the 'Datasets' options and click on maximize to view the window size with dynamically resized components.
- iv. Click on the 'Labeler' option and click on maximize to view the window size with dynamically resized components.
- v. Click on the 'Risk' option and click on maximize to view the window size with dynamically resized components.
- vi. Click on the 'Model Training' option and click on maximize to view the window size with dynamically resized components.
- vii. Click on the 'Predictions' option and click on maximize to view the window size with dynamically resized components.
- viii. Click on the 'Help' option and click on maximize to view the window size with dynamically resized components.

c. **Sponsor requirement:** Yes

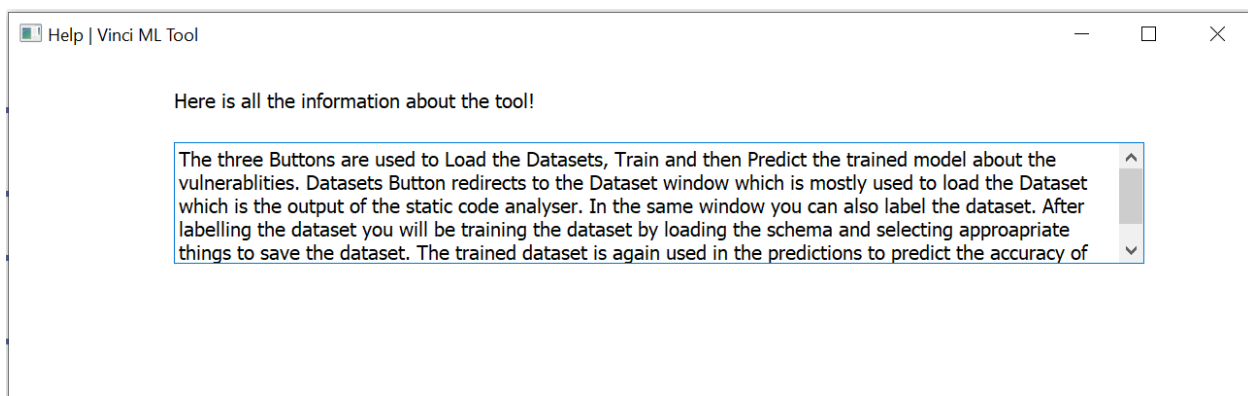
5. **Help option for each screen:**

a. **Description:**

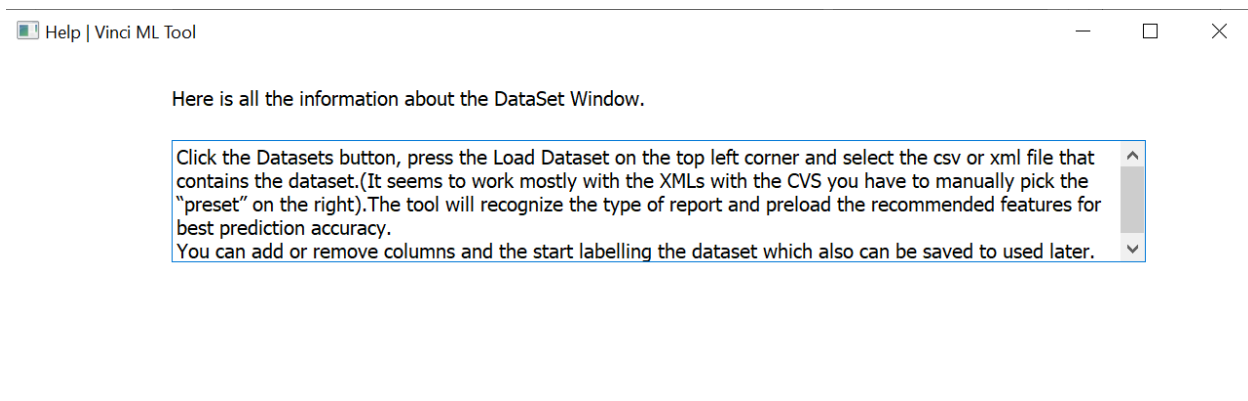
There is a lot of information in the tool which a user may or may not understand the fields and what they do. So, we decided on adding the help screen in every window which helps the user to identify the fields and work on the datasets in the tool. The help gives all the information about the windows and their functionalities.

b. **Feature user guide:**

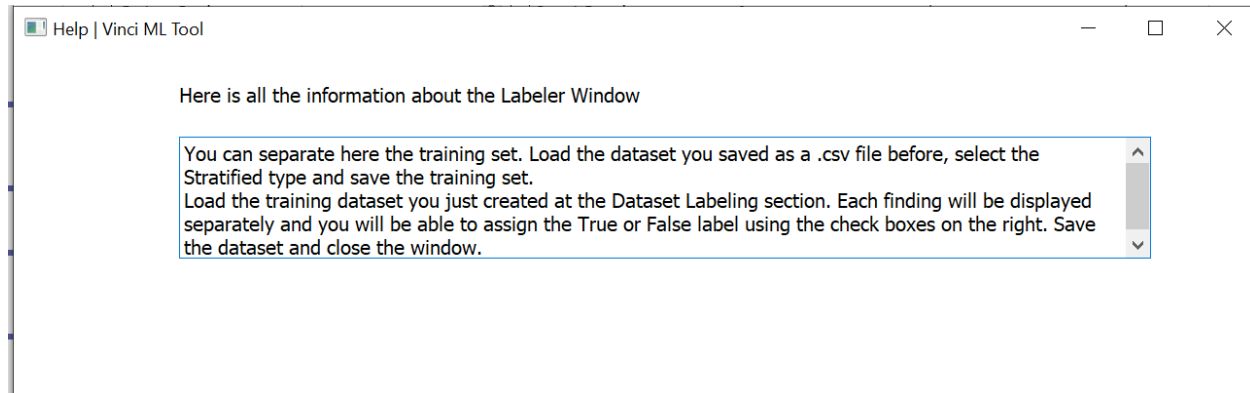
- i. Run the tool
- ii. Click on 'Help' to know about each of the features displayed



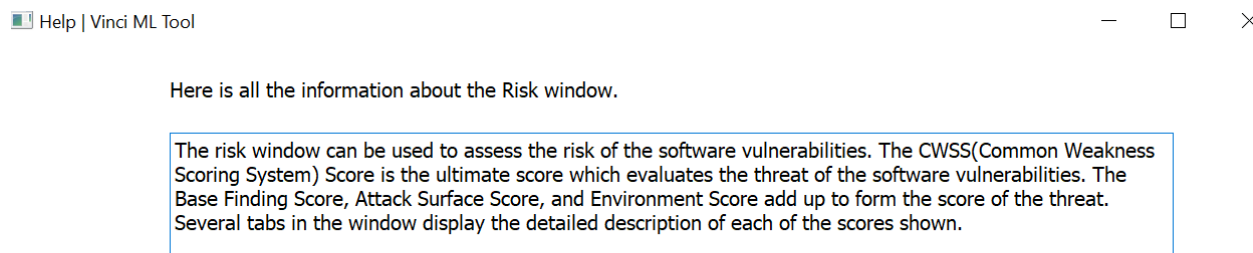
- iii. Click on 'Datasets' and click on 'Help' to know more about the 'Datasets' window and the functionalities involved.



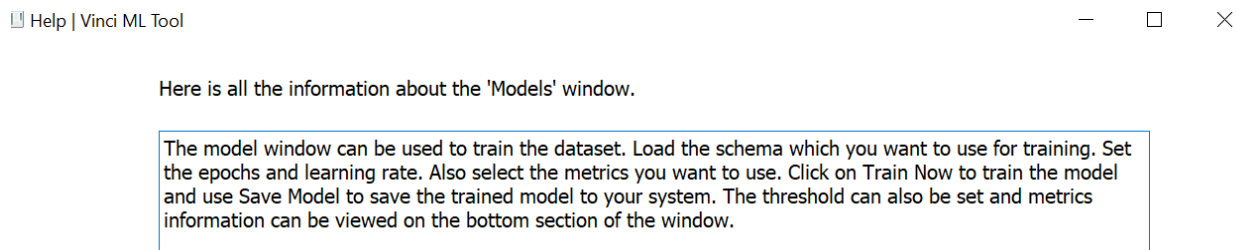
- iv. Click on 'Labeler' and click on 'Help' to know more about the 'Labeler' window and the functionalities involved.



- v. Click on 'Risk' and click on 'Help' to know more about the 'Risk' window and the functionalities involved.



- vi. Click on 'Model Training' and click on 'Help' to know more about the 'Model Training' window and the functionalities involved.



- vii. Click on 'Predictions' and click on 'Help' to know more about the 'Predictions' window and the functionalities involved.

Help | Vinci ML Tool

— □ ×

Here is all the information about the Prediction window.

Load the original dataset, the schema you created and the training model. Press Predict and Save Results when finished to save a .csv file with the predictions.

- c. **Sponsor requirement:** Yes

6. Tooltips for all screens:

a. Description:

- i. Tooltips are added to every label on all the available screens. This will help the user get an understanding of the functionality of that particular widget or textbox and helps enhance the overall UI of the tool.

b. Feature user guide:

- i. Run the tool.
- ii. Click on the 'Datasets' window.
- iii. Hover over any of the widgets or textboxes to display a tooltip.

Data:	<input type="text" value="Categori"/>	Training:	<input type="text" value="Yes"/>	Transformation:	<input type="text" value="One-hot"/>	Typ
Column		Use for training?		Data		Trar

Type of the data - Categorical(label values) or Numerical.

- iv. Click on the 'Labeler' button.
- v. Hover over a textbox in the window to display a tooltip.

Column Label:	<input type="text" value="output"/>	True Label:	<input type="text" value="True Positive"/>	False Label:	
Key		Name of the output column.	ue		

- vi. Navigate to the 'Risk' window and hover over a textbox to display a tooltip.
- vii. Navigate to the 'Model Training' window and hover a textbox to display a tooltip.
- viii. Navigate to the 'Prediction' window and hover a textbox to display a tooltip.
- c. **Sponsor requirement:** Yes

INSTALLATION GUIDE

The following steps are the guidelines to successfully install the project into the system and to create an executable for the same.

- **INSTALLATION:**

- Make sure you are using Python 3.8
- To install dependency packages, issue the following command from your project folder:

```
$ pip install -r requirements.txt
```

- After this, you should be able to launch the App with the following command:

```
$ python gui/app.py
```

- **CREATING THE EXECUTABLE FILE:**

- To generate an executable file, run the following command:

```
$ cd gui
```

```
$ pyinstaller --clean --onefile --windowed --icon=app.ico app.py
```

- To save executable to another folder, use --distpath:

```
$ pyinstaller --clean --onefile --windowed --icon=app.ico --distpath=distUbuntu/ app.py
```

ISSUES

There are no major pending issues in the tool. A few of the minor issues that could be improved in the tool are:

- The prediction made is less accurate so some of the Risk predictions made by the tool are wrong.
 - Possible Solution: This could be improved by using a different type of logistic regression modeling algorithm for risk prediction.
- The Risk Prediction can only be performed after the Labeler screen is visible. The Risk feature can be migrated to the Landing screen so that the user can perform the prediction without going to the Labeler screen.
- Some of the screens are redundant like modeling and prediction of Risk as well as vulnerabilities. This redundancy could be removed.
- Some of the UI screens follow different patterns than others. In the future, all the screens can be made in such a way that they follow the same pattern.

OUTLOOK

The tool can be further enhanced in the following ways:

- **Static code analyser tool support:**

Although the tool supports several static code analyser tools and their reports (such as PMD, PHP_Codesniffer, CodeDx, etc.), the support of a static code analyser tool for each

programming language would facilitate developers to choose Vinci tool to find the vulnerabilities in their software. Providing support for tools, such as Understand(for COBOL, FORTRAN, and PASCAL) will increase the reachability and usability of the tool.

- **Enhanced risk model:**

The current model for predicting the risk of software vulnerabilities has an accuracy of over 85%. However, enhancing the model will help increase the accuracy of the predictions and in addition, the several parameters used to calculate the risk can also be used to make the predictions as opposed to the use of only threat level in the current model.

- **Data transformation support:**

Currently the tool only supports the transformation of data from categorical to numerical using One Hot Encoding. Integration of the use of several other data transformation techniques will help in working with other types of data that are not compatible with the One Hot Encoding technique.

- **Remove redundant elements from GUI:**

The tool currently has a large number of screens and buttons which are quite redundant and unnecessary. Removing these redundant elements and screens will make the navigation seamless for the users and also improve the user experience. Screens currently do not flow sequentially. To access the screen for each functionality, the user needs to go back to the landing page and redirect to other screens from there on. Instead, the screens should follow a sequential order so that it is easy for first-time users to use the tool.

VIDEO PRESENTATION

- Youtube Link: [SER-517-Team-15-Final-Deliverable](#)