



International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Predictive Analytics for Crime Prevention by Using Machine Learning

¹Sk. Gouse Basha, ²A.V. Ramana, ³B. Murali Krishna

^{1,2,3}Department Master of Computer Applications, Rise Krishna Sai Prakasam Group of Institutions, Ongole

ABSTRACT:

Crime has emerged as a clear means of putting people and society in difficulty in our recent era. An increase in crime causes an imbalance in a nation's population. Understanding crime trends is essential for analysing and anticipating this kind of criminal activity. Using open source data from Kaggle, this study applies one such crime pattern analysis, which is then utilized to forecast the most recent incidents. This project's main goal is to determine which kind of crime contributes the most, along with when and where it occurred. In this study, some machine learning techniques, including Naive Bayes, are indicated.

KEY WORDS— Society, Anticipating, Criminal activity, Machine learning techniques.

Introduction:

Crime has emerged as a significant issue that is thought to be intensifying quickly. When an activity is against the law, is extremely offensive, and breaches a rule, it is considered to be a crime. The knowledge of criminology's various facets as well as the ability to identify patterns are both necessary for the crime pattern analysis. To use technology to control some of these illicit activities, the government has to put in a lot of time and effort. Therefore, it is necessary to employ machine learning techniques and its records in order to forecast the type and patterns of crime. It mandates the use of pre-existing crime data and makes predictions about the kind of crime that will occur and when based on the place and time. Numerous studies have been conducted by researchers that aid in the analysis of crime patterns and their relationships in a particular area. Some of the locations examined have made it simpler to categorise the crime patterns. This helps the authorities address them more quickly. This method makes use of a dataset from Kaggle open source that is based on several variables as well as the time and place where they occur during a predetermined period of time. I alluded to a classification system that aids in identifying the kind of crimes and concentrations of criminal activity that occur at specific times and days. A machine learning approach is proposed to be used in this case to match criminal patterns with their category using the provided temporal and spatial data.

EXISTING SYSTEM:

Pre-work is the process of removing duplicate values and features from a dataset that was collected from an open source. Decision trees have been used to extract features from massive amounts of data and to identify patterns in crime. It offers a basic framework for subsequent classification procedures. Deep neural networks are used to extract features from the identified crime patterns. The performance is computed for both trained and test values based on the prediction. The crime prediction aids in foreseeing future instances of any kind of criminal activity and assists the authorities in swiftly resolving them.

DISADVANTAGES OF EXISTING SYSTEM:

1. The pre-existing works account for low accuracy since the classifier uses a categorical value which produces a biased outcome for the nominal attributes with greater value.
2. The classification a technique does not suit for regions with inappropriate data and real valued attributes.
3. The value of the classifier must be tuned and hence there is a need of assigning an optimal value.

PROPOSED SYSTEM:

To remove redundant and irrelevant data values, the collected data is initially pre-processed using the machine learning techniques filter and wrapper. Additionally, it lowers the dimensionality; as a result, the data is now clean. After then, the data passes through another dividing step. It is divided into a trained data set and a test data set. Both the training and testing datasets are used to train the model. The next step is mapping. To make classification easier, the crime type, year, month, time, date, and location are all mapped to integers. Utilizing Nave Bayes, the independent relationship between the

qualities is initially investigated. The retrieved independent features are classified using Bernouille Nave Bayes. The labelling of the criminal aspects enables analysis of the incidence of crime at a specific time and area. The most common crimes are finally discovered, together with spatial and temporal data. The effectiveness of The accuracy rate is used to determine the prediction model. Python was used to create the prediction model, and the Co lab, an online compiler for data analysis and machine learning models, was used to run the programmed.

ADVANTAGES OF PROPOSED SYSTEM:

1. The proposed algorithm is well suited for the crime pattern detection since most of the featured attributes depends on the time and location.
2. It also overcomes the problem of analyzing independent effect of the attributes.
3. The initialization of optimal value is not required since it accounts for real valued, nominal value and also concern the region with insufficient information.

OVERVIEW OF SOFTWARE ENVIRONMENT PYTHON:

Python is a **high-level, interpreted, interactive** and **object-oriented scripting language**. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

HISTORY OF PYTHON:

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

PYTHON FEATURES

Python's features include:

- Easy-to-learn:** Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- Easy-to-read:** Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain:** Python's source code is fairly easy-to-maintain.
- A broad standard library:** Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode:** Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- Portable:** Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- Extendable:** You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- Databases:** Python provides interfaces to all major commercial databases.
- GUI Programming:** Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- Scalable:** Python provides a better structure and support for large programs than shell scripting.

PYTHON HAS A BIG LIST OF GOOD FEATURES:

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- IT supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

OVER VIEW OF SYSTEM TESTING:

Testing is done to look for mistakes. Testing is the process of looking for any flaws or weaknesses in a piece of work. It offers a means of examining the operation of parts, subassemblies, assemblies, and/or a finished product. It is the process of testing software to make sure that it satisfies user expectations and meets requirements without failing in an unacceptable way. Different test types exist. Every test type responds to a certain testing requirement.

TYPES OF TESTS:

UNIT TESTING:

Designing test cases for unit testing ensures that the core programme logic is working correctly and that programme inputs result in legitimate outputs. It is important to verify the internal code flow and all decision branches. It is the testing of the application's separate software components. Before integration, it is done following the completion of each individual unit. This is an invasive structural test that depends on understanding how it was built. Unit tests carry out fundamental tests at the component level and examine a particular configuration of a system, application, or business process. Unit tests make assurance that each distinct path of a business process adheres precisely to the stated specifications and has inputs and outputs that are well-defined.

INTEGRATION TESTING:

Software components that have been merged are tested in integration tests to see if they genuinely operate as a single programmed. Testing is event-driven and focuses more on the fundamental result of screens or fields. Even though the individual components were successful in unit testing, integration tests indicate that the combination of the components is accurate and consistent. Integration testing is especially designed to highlight issues that result from combining components.

FUNCTIONAL TEST:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation and user manuals.

SYSTEM TEST:

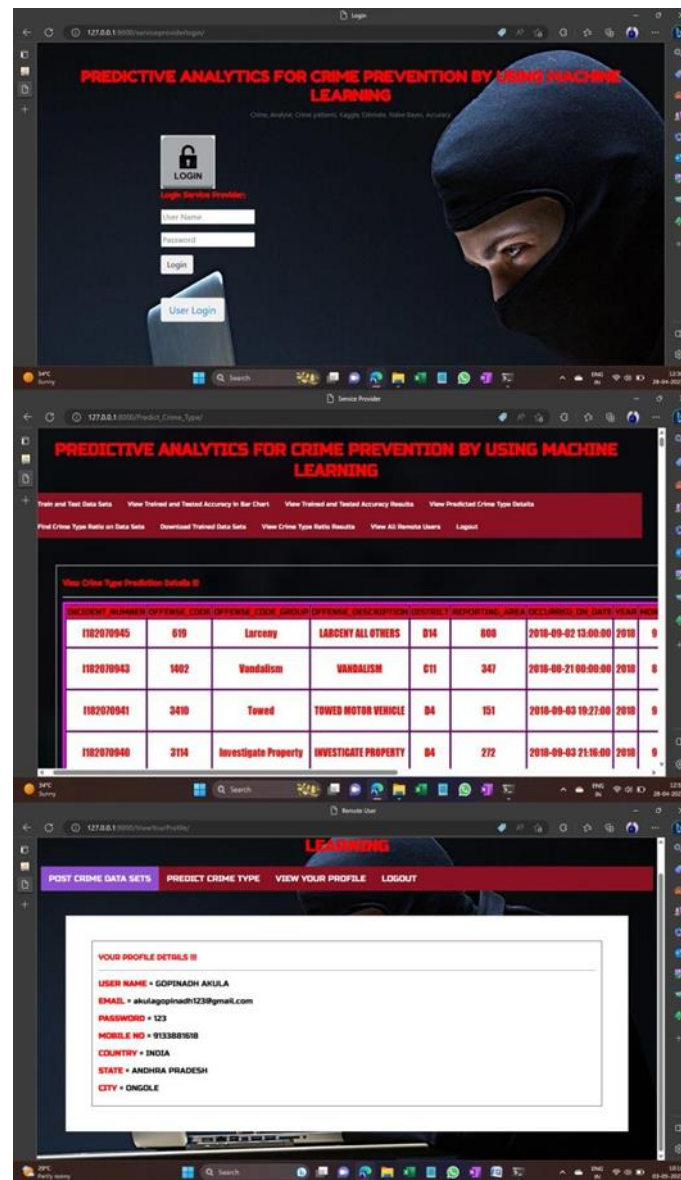
System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

WHITE BOX TESTING:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

BLACK BOX TESTING:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box.

RESULT:**CONCLUSION:**

In my project, I use the Multi nominal NB and Gaussian NB classifiers to help me cope with the nominal distribution and real valued characteristics.

Real-time forecasts can be made with the least amount of training time possible. Additionally, it solves the issue of dealing with a continuous target set of variables, which the previous work was unable to accommodate. Therefore, Nave Bayesian Classification could be used to forecast and identify the crimes that occur the most frequently. Additionally, the algorithm's performance is determined using a few common criteria. The metrics average precision, recall, F1 score, and accuracy are the ones that matter most when evaluating an algorithm. The use of machine learning algorithms could significantly improve the accuracy value.

REFERENCES:

1. Suhong Kim, Param Joshi, Parminder Singh Kalsi, Pooya Taheri, "Crime Analysis Through Machine Learning", IEEE Transactions on November 2018.
2. Benjamin Fredrick David. H and A.Suruliandi, "Survey on Crime Analysis and Prediction using Data mining techniques", ICTACT Journal on Soft Computing on April 2012.

3. Shruti S.Gosavi and Shraddha S.Kavathekar, "A Survey on Crime Occurrence Detection and prediction Techniques", International Journal of Management, Technology And Engineering , Volume 8, Issue XII, December 2018.
4. Chandy, Abraham, "Smart resource usage prediction using cloud computing for massive data processing systems" Journal of Information Technology 1, no. 02 (2019):108-118.
5. Learning Rohit Patil, Muzamil Kacchi, Pranali Gavali and Komal Pimparia, "Crime Pattern Detection, Analysis & Prediction using Machine", International Research Journal of Engineering and Technology, (IRJET) e-ISSN: 2395-0056, Volume: 07, Issue: 06, June 2020
6. Umair Muneer Butt, Sukumar Letchmunan, Fadratul Hafinaz Hassan, Mubashir Ali, Anees Baqir and Hafiz Husnain Raza Sherazi, "Spatio-Temporal Crime Hotspot Detection and Prediction: A Systematic Literature Review", IEEE Transactions on September 2020.
7. Nasiri, Zakikhani, Kimiya and Tarek Zayed, "A failure prediction model for corrosion in gas transmission pipelines", Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, (2020).