SPATIAL CRIME ANALYSIS AND PREDICTION

ENROLLMENT NO: 16803006, 16803008, 16803029

NAME OF STUDENTS: SHUBHAM AGGARWAL, SWARAJ SAXENA, PARINAY PRATEEK

SUPERVISOR: DR. MANISH KUMAR THAKUR

MAY - 2020

Submitted in partial fulfillment of the Degree of

5 Year Dual Degree Programme B. Tech

in

Computer Science Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

TABLE OF CONTENTS

(II)

DECLARATION

I / We hereby declare that this submission is my / our own work and that it contains, to the best of my knowledge and belief, no material previously published or written by another person or material approved for the award of any other degree or diploma from the university or other institution of higher learning, except where due recognition has been granted in the document.

Signature:	...…..…………………..	….……………………	…………..……….
Name:	**Shubham Aggarwal**	**Swaraj Saxena**	**Parinay Prateek**
Enrollment No:	**16803006**	**16803008**	**16803029**

Date: ……………………

Place:

(III)

CERTIFICATE

This is to certify the work is titled "**Spatial Crime Analysis and Prediction**" submitted by "**Shubham Aggarwal (16803006), Swaraj Saxena (16803008), Parinay Prateek (16803029)**" in partial fulfillment for the award of degree of **5 Year Dual Degree Programme** of **Jaypee Institute of Information Technology, Noida** Was done under my supervision. This work was not submitted to any other university or institution in whole or in part for the award of this or any other degree or diploma.

Signature of Supervisor	……………………..
Name of Supervisor	**Dr. Manish Kumar Thakur**
Designation	**Associate Professor**
Date	……………………..

(IV)

ACKNOWLEDGEMENT

We would like to extend a heartfelt thanks to our college "**Jaypee Institute of Information Technology, Noida"** for giving us an opportunity to work on the project "**Spatial Crime Analysis and Prediction**".

The successful completion of this project would not have been possible without the support of our Supervisor **Dr. Manish Kumar Thakur**. In every phase of this project his support and guidance was very valuable for us.

We also want to thank our friends for supporting us and giving their suggestions during the development of this project.

Signature:	...…..…………………..	….……………………	…………..……….
Name:	**Shubham Aggarwal**	**Swaraj Saxena**	**Parinay Prateek**
Enrollment No:	**16803006**	**16803008**	**16803029**

Date: ……………………..

(V)

SUMMARY

Crime is a sort of human social interaction, and crime prediction is a part of the social event prediction debate. In recent years work has focused on predicting the occurrence of large-scale social events. These are all observational studies, however, which only offers minimal assistance.

Governments expend plenty of money trying to deter crimes from occurring by law enforcement agencies. Today, several law enforcement agencies have vast amounts of crime-related data, which have to be analyzed to translate into usable information. Crime data is complicated as it has several dimensions and includes string records and narrative records in various formats, e.g. most.Because of this variety, using off-shelf, computational, and machine learning data analytics tools find it difficult to mine. It's the prime explanation why crime data mining lacks a general forum.Although there are some proprietary frameworks for predicting and analyzing crime stats, they concentrate only on specific, non-extensible areas of crime and do not include an API for integration with other tools.

We did a lot of research to incorporate the Crime Prediction Model in the beginning but most of the work does not achieve much precision and effectiveness. Later we found that CNN outperforms all other models in the extraction of features, and also that LSTM is the best algorithm that addresses long-term dependencies. Therefore we combined the algorithms which formerly known as ConvLSTM for evaluating and predicting crime.

The results obtained ranges from 80-85 percent(approx.) accurate.

…..…..…………………..　　　…..…………………　　　……………..……….

Shubham Aggarwal　　　**Swaraj Saxena**　　　**Parinay Prateek**

………………………………

Dr. Manish Kumar Thakur

(VI)

LIST OF FIGURES

LIST OF TABLES

CHAPTER -1

INTRODUCTION

1.1 General Introduction

Crime is like a human social interaction, and crime prediction is a part of the social event prediction debate. In recent history work has focused on predicting the occurrence of wide-scale social events. These are all observational studies, however, that only offer minimal assistance.

Crime prediction is a major issue that has to do with social security and the survival of people. This is more appealing from the perspective of ordinary people, as citizens or visitors, to tell them the outcome of a qualitative prediction than a quantitative one.But it is much more realistic to consider the quantitative forecasting findings, which are the most important basis of an intelligent decision, in the view of police deployment, or decision maker.

As the third largest city in the USA, Chicago is known not only in film and television works but also in the real world for its high crime rate. It is a city that draws a great deal of publicity for fighting crime. Anyhow, these researches often concentrate on anticipating a specific form of crime.Though, the task of estimating when and where the crimes will occur is extremely challenging because it is prone to the extremely dynamic spatial and time distributions of crimes. Whereas recent research recognized that events involving crime continue to show spatial and temporal dependency on the complex social environments. Spatial dependencies indicate a region's crime risks being influenced by the crime-related incidents or environmental factors in its surrounding spatial regions as well as far away regions. For example, empirical research established that the spatial concentration of crime was statistically related to burglary victims, bars, wages, race populations and traffickings.On the other hand, the temporal dependencies mean that a region's crime risks are determined at recent, close and even distant time intervals by the crime-related events or environmental factors. For example, the close-repeated trends found in crimes suggested that the recent consistently committed crimes are seen as the important variable for predicting the risks of local crime in the immediate future.In addition, complex social activities such as 311 or 911 accidents, in the near-spatio temporal setting, may mean high crime risks. In summary, the crime prediction task centered on determining the successful spatiotemporal dependencies from the complex interplay of space-, time- and environmental variables.

Countries expend plenty of money trying to deter crimes from occurring by law enforcement agencies. Today, several law enforcement agencies have vast amounts of crime-related data, which need to be analyzed to translate into usable information. Crime data is complicated as it has several dimensions and includes string records and narrative records in various formats, e.g. most.Because of this variety, using off-shelf, computational, and machine learning data analytics tools find it difficult to mine them. It is the primary explanation why crime data mining lacks a general forum. Although there are some propitiatory tools for forecasting and analyzing crime data, they concentrate only on specific, non-extensible areas of crime and do not have an API for integration with other tools.In addition, it is not possible to use the same method for the study and preparation, such as patrol beads and district borders.

1.2 Problem Statement

The research question this project aims to tackle can be described as follows: How to build a software framework to perform descriptive, predictive and prescriptive analysis of various crime data??

1.3 Significance / Novelty of the problem

The objective of the project aims in analyzing and predicting the crime and crime hotspot spread in the city of Chicago, USA. There are various benefits of predicting something disastrous which will be going to happen in future so as to medicate the effect of its happening:

1. Help prevent recurring crimes in an area by tracking the patterns of crimes as well as the most common types of crime in an area.
2. Early warning systems and constant vigilance through police patrolling can be employed in hot spots of crime.
3. Consistency in record keeping, analyzing what type of force is needed, the number of encounters and fatalities can be collected using crime forecasting.

In recent years, work has concentrated on qualitative predictions of a particular form case, such as riots or gun crimes. However, incidents such as crimes typically have complex associations with each other, and predictions of a particular form of case can not match the real demands. Hence, the model helps in predicting crime which thereby helps the Police to prevent its happening or take some action accordingly. Also, the model can easily be integrated to the Chicago Govt. Crime Department and this will help the police to arrange beats in those red zone areas so as to medicate its effect. The accuracy of the model is the biggest novelty to our problem statement.

1.4 Empirical Study

CHICAGO Crime Dataset - The Chicago crime dataset comprises 70 Lakhs approx. crime entries from the year 2001 to present which includes more than 10 different crime types like theft, homicide, criminal damage, assault,robbery ,narcotics, weapon violation, burglary, deceptive practices elc. Each of the crime data entries consists of Date, time, Address of the crime Location, Crime type, Case Number, Latitude & Longitude,Police Beat. The dataset used for the research is accurate, actual and authentic since data is collected from the Chicago's Crime Portal official website. Some of the entries have lat & long missing which causes disruption in the training process. We have filled some of the entries using Google Geospatial Finder but due to software restriction, we have to remove most of the entries having the same issues. This dataset has been frequently used by the research community as the crime data is easily available at the Chicago Crime Portal. We have used Monthly Crime Data from the year 2012 to 2014 for training and 2015 Monthly crime data will be used for validation and testing.

Tensorflow - TensorFlow is a free and open-source software library for dataflow and differentiable programming through a range of functions. It is a library of symbolic math, which is also used in machine learning applications such as neural networks. It's used on Google for both research and production. TensorFlow is a second-generation framework for Google Brain. While the implementing reference runs on single machines, TensorFlow can run on multiple CPUs and GPUs. TensorFlow is available on Linux, Windows, macOS and 64-bit. Its flexible architecture allows computation to be easily deployed across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to server clusters to mobile and edge devices.

Google Colab - Colaboratory is Google's free Jupyter Notebook environment where you can use free GPUs and TPUs to solve all of these issues. Of course there are limits. It supports Python 2.7 and 3.6 but still does not support R or Scala. There is a limit on your sessions and duration, but if you're imaginative and don't mind re-uploading your files regularly you can certainly get around that. For everything from developing your Python coding skills to collaborating with deep learning libraries such as PyTorch, Keras, TensorFlow, and OpenCV, Colab is perfect. You can build Colab notebooks, upload notebooks, store notebooks, exchange notebooks, mount your Google Drive and use everything you have saved in it, import most of your favorite folders, upload your personal Jupyter Notebooks, upload your notebooks directly from GitHub, upload Kaggle files, access your notebooks, and do whatever else you may like to do.

1.5 Brief Description of the Solution Approach

It can be very well observed that a lot of research has already been done in this field and a lot is yet to be done. To achieve the goal of this project, initial steps were to analyse what are the previous models that had already projected their work on Crime prediction. After analysing and researching different ML models, it can be concluded that for feature extraction, CNN showed unparalleled performance among all the models. For prediction, LSTM outperforms all other models as it considers long term dependency which is required for time series forecasting models. Hence, we used a combination of both the models (CNN+LSTM) which formerly known as ConvLSTM.

The image passes through the convolutionary layers in this Algorithm and the result is a set flattened to a 1D array with the features obtained. The effect is a set of features over time, and this is the input of the LSTM layer, while performing this process on all images in the time series.

The architecture begins with four Conv2D layers, each accompanied by a Normalization of the Array. It splits up into divisions in sequence, one for each group. Then this output is connected to a Dense network which is completely connected. At last, the very last layer is a single-cell Dense.

Then, 4 ConvLSTM2D models are implemented each followed by Batch Normalization. After all the training is over, we receive a shape file of the crime hotspot of the next month and then the predicted image is compared with the actual crime hotspot of that month and accuracy is checked.

Also, the model is compared with the existing Machine Learning model like Decision tree, linear SVM Classifier, PCA, KNN, Naive Bayes Classifier with the same Crime dataset. It can be clearly proved that our ConvLSTM model outperforms all the models both in terms of Training time and Accuracy.

1.6 Comparison of the Existing Approaches to the Problem Framed

Since we have gone through different research work on forecasting and crime prediction algorithms / model, we have found that there are numerous models, such as Support Vector Machine (SVM), which is a linear model and is already being implemented for time series prediction, classification tasks, and regression[21].The output of the SVM is contrasted with the success in time series prediction of both the recurrent neural network model (RNN) and the autoregressive moving average (ARMA)[22]. ARMA model is a linear system but it is simple and fast to use while RNN can be a linear model in theory but is typically difficult to train and does not lead to a single or global solution due to variations in its initial weight set[22].Previous methods for forecasting, such as Auto-Regressive Moving Average (ARMA)[23] and Support Machine (SVM)[24], assume a constant temporal time series pattern, which can be restricted. Furthermore, provided that only recent data are considered.A lot of valuable knowledge (e.g. long-term consequences of transient associations of crimes) would be lost to make forecasts and past instances down-weighted, reducing the already scarce crime data and the non-linear and stochastic nature of crime pattern these models not being able describe unique nature well and result in bigger prediction errors.

According to [25], an **artificial neural network** (ANN) is a model inspired by a biological neuron and acts as the human brain in decision making. It consists of a large number of processing components that work together to solve problems [25]. Meanwhile, ANN model is based on the prediction by smartly analyzing the trend from an already existing voluminous historical set of data [26]. ANN has more flexible and general functional forms and can effectively deal with them than the traditional statistical methods [27]. Frequently, traditional statistical prediction models have limitations functions to estimate the relationship between the inputs and the outputs (the future values) due to the complexity of the real system but ANN can be used as an alternative method to resolve these limitations functions [28].

According to [29], Learning the **decision tree** uses a decision tree as a predictive model that maps observations on an item (represented in the branches) to conclusions about the target value of the item (represented in the leaves). It is one of the predictive modeling methods used in analytics, machine learning , and data mining. **SVM** are supervised learning models with associated learning algorithms which analyze the data used to analyze classification and regression. In the light of a set of training examples, each of which is classified as belonging to one or the other of two categories, an SVM training algorithm generates a model that assigns new examples to one or the other category, making it a non binary classifier. In a learning problem, **Naive Bayes classifiers** are highly scalable, requiring a number of linear parameters in the number of variables (features / predictors). Maximum likelihood training can be achieved by evaluating a closed-form expression that takes linear time rather than costly iterative approximation as used in many other classifier forms. Principal Component Analysis (PCA) is one of the most commonly used multivariate data analysis and it can be considered as a projection approach that projects observations from the initial dimensions from a p-dimensional space with p-variable to a k-dimensional space [30].

After having a deep research on crime analysis and prediction we implemented convLSTM Neural Network model in which features extraction is done by CNN and long term dependencies between crimes over a period of time done by LSTM. CNN is best amongst existing features extraction models and for time series prediction and finding long term dependencies LSTM is the best as far as our research is concerned.

CHAPTER -2

LITERATURE SURVEY

Section 2.1 defines crimes and related theories. Section 2.2 discusses the Summary of the research paper studied and section 2.3 discusses the integrated summary of literature reviewed.

2.1 Crimes

2.1.1 Crimes and Social consequences

A crime can be defined as any action or omission that contravenes a law, resulting in a punishment. What constitutes a crime typically depends on the bodies of government and on the rules that exist in those areas. To understand the nature of crimes, it is important to consider not only their spatial-temporal aspects, but also the essence of the crime, the relationship between victim and perpetrator, the involvement of the guardian, and record of similar occurrences [8]. Regardless of why crimes are committed, they are putting a strain on the communities, towns and cities. Usual financial costs associated with them include the expense of fighting investigation and prosecuting criminals. Non-monetary costs are social costs, impacting the quality of life, mental health, and physical protection of people living in those regions. Crimes are a social annoyance and being able to solve them more quickly is extremely important and will pay for itself [4].

2.1.2 Criminology Theories

According to John and David[9], crime theories may be split into two groups, namely those seeking to understand the evolution of criminal suspects and those seeking to explain the evolution of criminal incidents. Criminology was built primarily through perpetrator hypotheses and research. It has only recently started to clarify the actions of the persons involved, rather than the criminality. Criminology is made up of many theories which explain how and why some offenders act in the way they do. Following are some theories that explain how places are associated with crimes [9].

1. Rational Choice means criminals might choose strategies and identify ways to reach their goals in a manner that can be understood. It can also be described as human actions are based on logical judgments, i.e. they are guided by the possible consequences of this action [9].

2. Routine Activity Theory This theory describes how crimes arise as a result of numerous circumstances. In other words, a motivated offender, a desirable objective, objective and offender must be present simultaneously and, last but not least, must be absent from other forms of intimate controllers, guardians and place managers [9].

3. Crime Pattern Theory This theory incorporates the above two theories and goes on to suggest that the manner in which targets enter offenders' attention is determined by the distribution of crime events over time, space, and between targets. An individual may become aware of criminal activity when engaged in their normal day-to-day jobs. Therefore a given person would only be informed of a subset of goals available. The definition of location is crucial to the theory of crime patterns[9]. Understanding the theories of criminology is important in attempting to build resources or outlets for investigating crime using new technologies.

2.1.3 Crime Analysis

Reporting of crime is a challenging task as it involves both the compilation and review of huge amounts of information. For example, Brown[6] notes that there are roughly 100,000 criminal records each year in Richmond city in the USA. Given the volume of data, and the need to apply various algorithmic techniques, manual analysis is prohibited. Whereas an automated analysis of such a rich collection of data could detect complex trends of crime and help to solve crimes more quickly.

Data mining techniques can be used for analyzing crime data, criminal career analysis, analyzing bank fraud and analyzing other important problems in law and enforcement[8]. Some of the conventional techniques of data mining are association analysis, classification and prediction, cluster analysis and outlier analysis, which recognize trends in structured data[10]. Using theories of criminology combined with modern technology will help to easily and efficiently recognise trends of crime. A crime data analytics tool may be used to automate the workload and would help simplify the process while providing more accurate and informative conclusions and predictions.

2.2 Description of the papers reviewed:

They explained the estimate of kernel density with GI * hotspot analysis, and applied the techniques in the crime dataset, according to [2]. KDE aims at detecting high value clusters within the data while GI * not only detects but also deepens the perception of spatial clusters. For point position, they use kernel density to measure the density of point features around each output raster cell. Thus they concluded by jointly applying both kernel density and hotspot analysis for better understanding. The downside was that they do not recognize factors such as area, population, POI locations that have a significant impact on the events of crime.

They introduced predictive hotspot mapping, in line with [8]. The KDE takes no account of the temporal dimension of the crime. They compare previous models and find 4 characteristics, and add estimation of spatio-temporal kernel density to predictive hotspot mapping. They then use cross-validation to detect appropriate bandwidth and then filter positive hotspots and apply a predictive accuracy index to calculate accuracy of the model implemented. They do not find that a significant factor, i.e. Event time, and the temporal element of crime as well. I also find only one form of crime that isn't possible in the real world.

They gathered data from British Columbia, Canada, according to Spatial[9]. They applied predictive modeling and picked 4 characteristics: financial, spatial, geo-social and similar. They applied their model to the co-offender activities and created a co-offending G (V, E) network. Increasing nodes represents a known offender and when 2 nodes are linked, at least one crime is committed. They were implementing this model on 4 techniques of classification: Naïve Bayes, Random Forest and Bagging. The downside was that they did not find more details about criminals, and they also did not recognize trends of non-co-offending to minimize crime.

They collected 15-year Vancouver crime data, and then pre-processed it to fill the empty cell, redundant columns, etc., according to [10]. They then carried out pattern analysis by making a heat map of the crime data every day and every month. This transforms all categorical variables into binary values of 0 and 1. All the days were made into features and then KNN was added and Boosted decision tree to predict the next result. The paper's downside was that the precision was very poor and their inability to predict nonlinear dependencies. Therefore, the Artificial Neural Network was implemented with development to take care of non-linear dependencies and improve accuracy.

They initially shape crime hotspots that are plotted on the map according to [11]. They then assign different weights based on clustering for different types of crimes. The paper essentially includes a Detective who will research these clusters and will help identify the person or suspect according to his previous records. This paper essentially aims to identify all those people / gangs who might be the culprits for the actual incident that took place. The downside was that they would have to recruit and integrate a researcher who would research these clusters and would also have an effect on the clusters being identified by adding a new suspect.

They developed 3 frameworks according to[12]. The first model was Deep CNN for predicting end to end crime. Later, they combine inception and fractal network so that it learns dependencies between crimes automatically. They then measured STCN using New York's Crime dataset and compared their success with four base models. Break the model into a grid and apply the STCN model to that grid. The downside of this paper was that for crime prediction they did not consider types of crime, and also various geo factors such as population, location, POI.

Two traditional methods-temporal and empirical models-were introduced as per [13]. The model takes an image file as input, and then forms a 100 * 100 matrix grid for that image. Every grid cell shall have a value representing the value of a feature. The grid is then applied to various machine learning models, such as SVM, Random forests, decision tree and their proposed "DNN Tuning" algorithm. All other models are done by DNN tuning out. The downside of this paper was that they did not include displacement features such as buffer zones, police station locations, Public parks. That may have an impact on new crime events.

We lay stress on the definition of interwoven time series, according to [5]. Because different time series have distinct time intervals and their own pattern, we have made each time interval as a completely linked structure up to the last hidden layer. A completely connected system with a minimum interval is built into the first sheet. Then the information is collected for each hidden layer until the last hidden layer and then combine the data from all intervals to get the information layer. They aggregate all forms of crime into data, and then forecast the crime for 10 days to come. The downside was that they did not take into account long-term dependencies.

According to [14], they made crime hotspots first of all from data obtained from the events of the crime. The features of the hotspot with more crime events are extracted and transferred to the RNN model and forecast the crime by separating the layers and creating fully linked systems and collecting results from each layer and combining all the data to get the output layer. The downside was that they overlooked an significant aspect, i.e. external parameters are taken as constant that doesn't affect the crime that isn't possible in real life.

They gathered data from the cities of Chicago and Portland, according to [16]. Every city is divided into grid cells (Chicago beast, and Portland square grid). For one day every entry has some features. They wanted to estimate the next day's crime count. They just find data on a few forms of crime. They assign weight to the network, randomly and to predict performance perform forward pass with current weight. They also calculated loss function which is the difference between true and expected output and using this they calculated gradient using derivative function, and then average all records and continue this process for 10 epochs. They then passed the features to 4 models, and compared these models' accuracy. Their downside was that they didn't recognize long-term dependencies that could improve their accuracy. They also find only a few forms of crime which are not only forms of crime in this present world.

They also established a new system for crime prediction according to [17][20]– Deep Crime Network. In particular, they first developed a vector that represented the crime matrix frequency, the point of interest matrix, and the matrix of urban anomalies. Their system has a three-level architecture of Gated Recurrent Units (GRU) to encode the temporal dependencies of the sequence of crime, anomaly sequence and their inter-dependencies, respectively. In order to model the underlying region-category interactions, we first add an input weight vector for each region to differentiate the occurrences of which past crime categories are more important for future predictions. Then, they concatenate the embedding vector area and crime and feed it into a Multilayer Perceptron Layer.

The raw data from the NSE site was obtained and processed using 4 stages, according to [6]: the first stage was data discretion, then data transformation, then data cleaning and then data integration. After that the collection of data is split into training and research. Features which are date, high, low, open, close, are extracted. These extract features have been fed to the LSTM model which consists of a sequential input layer followed by 2 layers of LSTM and a dense layer will activate ReLu which results in linear activation function. The generated output value is then compared with the target value. We introduced the stock prediction to understand LSTM's functionality and role.

We investigated the ability of DL methods to forecast hotspot areas in an urban setting, where crimes of certain types are more likely to occur in a given future window, according to[18]. To achieve this goal, we fed the DL methods with the minimum amount of data containing only details of the spatial, temporal and type of crime. We used a dual output environment where the second output is the number of crimes that happened in the same future window in order for the models to accurately predict the order of "hotness" In addition, we picked our SFTT model configuration as the winning one compared to 10 different algorithms in 5 crime incidence datasets, and further evaluated the parameters picked for robust performance.

They applied the STARMA model, according to [19], by taking a set of crime data, and then applying auto-correlation and smoothness techniques. The performance is then verified using patterns, and LSTM is then applied to this model to measure the predictive results.If the Smoothness performance is a random set of data, then using estimation parameters and predicting the outcome, they apply the STARMA model by defining a weight matrix and then applying a STARMA recognition model. The results were then verified using MAE, RMSE techniques for STARMA and hybrid (STARMA+LSTM) and found that it improves the non-linear functionality of the model as well. The consequence of this report was that the number of crimes per month had increased. The downside of this paper was that the matrix of weight is constructed using the method of center distance which is the worst technique for measuring spatial distance. Also, missing entries or invalid entries may cause the STARMA model to be applied and, as data is not in a continuous sequence, precision will be reduced.

They implemented the model on traffic flow data from the past 30 minutes and at a series of 6 data points to predict traffic flow from next 5 minutes, according to[7]. Data is broken down into two parts: 3 weeks to train and last week to test. The features vectors are then input into the LSTM+GRU model and estimate the flow of traffic. They have compared their findings with other model baselines, such as ARIMA using MSE and MAPE. We reviewed this paper to understand the workings of LSTM and GRU and find that compared to the ARIMA model, it has reduced The MAE to around 10 percent. The downside is that they only used LSTM and GRU for short term dependencies.

2.3 Integrated review of the examined literature:

Over the course of the study, we incorporated the review of all the research papers that we read. Any of the articles summaries were cited below:

S No.	Name	Algorithm Implemented	Summary
1	**A Spatio-Temporal Kernel Density** **Estimation framework for predictive crime hotspot mapping and evaluation**	● Kernel Density Estimation ● Density Estimation To estimate the density of an event, the STKDE developed by Brunsdon et al (2007) multiplies a bivariate kernel placed over the x-y domain with	Experts and law enforcement officials are well aware that crime appears to be concentrated in some areas (e.g., Bernasco, Johnson, & Ruiter, 2015; Bowers, Johnson, & Pease, 2004; Chainey, Tompson, & Uhlig, 2008); The first form of approach aggregates cases of crime to be counted and measures rates by geographical boundaries such as units of censuses It uses the multivariate regression model to research the

		a univariate kernel along the temporal dimension t. They used it to identify and imagine trends of crimes.	relationship between crime levels and a wide variety of attraction and inhibition variables, such as socioeconomic factors, community demographics, forms of land use, cultural values, and history of drug abuse (e.g., Bushman, Wang, & Anderson, 2005; Kikuchi & Desmond, 2010; Peterson & Krivo, 2010). Predictive hotspot mapping is commonly used by large law enforcement agencies, especially large ones serving over 500,000 U.S. citizens (Hart & Zandbergen, 2014; Reaves, 2010)
2	**An Interweaved Time Series Locally Connected Recurrent Neural Network Model on Crime Forecasting**	● Recurrent Neural Network ● Time Series In this paper they researched RNN structure quantitative crime forecasting. They suggest an interweaved time series approach for dealing with time series at various intervals in order to hold the patterns of data fluctuation in different perspectives. They then implement this method in two common RNN models, LSTM and GRU, where different intervals are interpreted and treated using a locally connected structure. In studies they use actual open	They use the GARCH model as the baseline. Completely linked LSTM and GRU are also chosen as the comparison models for checking the validity of the ILC-RNN structure. Mean Squared Error (MSE, Eq (4.1)), Root Mean Squared Error (RMSE, Eq (4.2))) or Mean Absolute Error (MAE, Eq (4.3)) was selected for evaluation. To better explain the result's characteristics, our preference is both MSE and MAE. In their tests, our finding is that data sequences such as pred3 have more data points similar to the actual results, but the outliers have bigger errors, whereas sequences such as pred5 are smoother but with smaller values get more deviation points. From the standpoint of their problem of forecasting, between the two predictive findings there is no completely better one. Yet in the view of the decision maker, a better option should be made, based on the particular circumstances.

		source data from the Chicago Police Department to make comparisons between their approaches and the three reference models.	
3	**Kernel Density Estimation (KDE) vs. Hot-Spot Analysis - Detecting Criminal Hot Spots in the City of San Francisco**	● Kernel Density Estimation ● Density Estimation ● Point Pattern Estimation ● Point Pattern Analysis They explore additional methods of evaluating point trends, such as evaluating the density or statistical operations with different refinements and extensions. Hence, in this paper they concentrate on two commonly used forms of analysis of point patterns – kernel density analysis (KDE) and hot spot analysis of Gi*. They conduct both types of analyzes on the same dataset-San Francisco criminal data and evaluate their output based on data	Point pattern analysis occurs in several different areas of study (Cressie, 2015). A point process is, in general, a stochastic process in which they determine the positions of some important events within a bounded area (Bivand et al, 2008). Point pattern analysis includes the ability to analyze and explain patterns, and check whether a random spatial pattern has a substantial difference (O'Sullivan and Unwin, 2003). As such it represents one of the most important geographical and spatial theoretical principles. In the case of KDE, a general recommendation regarding parameter settings is difficult to provide, because they are based on user requirements. Knowledge described by the resulting surface density depends on the choice of kernel bandwidth and the size of the output grid. Therefore it is important to play with these parameters in order to obtain a map suited to the needs of the consumer. The hotspots originating from the KDE map are not statistically important, and the results may obviously be influenced by various cell sizes and search bands. In these cases, patients should be careful regarding the field of diagnosis, the field of research and the case of research. On the other hand, they can estimate the density distribution of events at local level with hot spot analysis, and find statistically important hot spots within

		characteristics and case study target	their dataset.
4	**Deep Convolutional Neural Networks for Spatiotemporal**	- Convolutional Neural Networks - Support Vector Machines The input function maps of the crime and 311 data are translated to two 2D image-like arrays. The model is able to capture the low-level spatiotemporal dependencies for crime events by moving them into a series of two convolution layers. As the output function maps flowing deep in the networks, the model began to abstract the high-level spatiotemporal features mainly relying on inception blocks and fractal blocks, which utilize branches of convolutional layers and merge layers to combine the crime-related features with different abstract rates. Finally, in order to	In this paper, we proposed a novel model for crime prediction based on the deep CNNs. Comparing 4 crime-based baseline methods and 311 results, the proposed model demonstrated its superiority in the task of predicting spatiotemporal crime risks. However, CNN is a "black-box," in which the mechanisms of the neuron relation are not readily interpretable. While this report, rather than waiting for the day when the "black box" is open, is eager to take a fast pace to apply deep CNNs to the analysis of spatiotemporal crimes to deter crime now. We assume that this research only scratches the surface of what is possible in this direction and there are several avenues for further study such as modeling various types of risk of crime, defining significantly socio-economic characteristics for crime prevention etc. In addition, more types of data should be considered in order to improve potential accuracy of predictions.

		realize the crime risk prediction, the highest-level crime-related features were aggregated into the dense layer which acts as the classifier within the system.	
5	**A Spatio-Temporal Kernel Density Estimation framework for predictive crime hotspot mapping and evaluation**	● Kernel Density Estimation ● Density Estimation To estimate the density of a case, the STKDE built by Brunsdon et al (2007) multiplies a bivariate kernel placed over the x-y domain with a univariate kernel along the temporal t axis. They used it to identify and imagine trends of crimes	Experts and law enforcement officials are well aware that crime appears to be concentrated in some areas (e.g., Bernasco, Johnson, & Ruiter, 2015; Bowers, Johnson, & Pease, 2004; Chainey, Tompson, & Uhlig, 2008); The first form of approach aggregates cases of crime to be counted and measures rates by geographical boundaries such as units of censuses It uses the multivariate regression model to research the relationship between crime levels and a wide variety of attraction and inhibition variables, such as socioeconomic factors, community demographics, styles of land use, cultural values, and history of drug abuse (e.g., Bushman, Wang, & Anderson, 2005; Kikuchi & Desmond, 2010; Peterson & Krivo, 2010). Predictive hotspot mapping is commonly used by large law enforcement agencies, especially large ones serving over 500,000 U.S. citizens (Hart & Zandbergen, 2014; Reaves, 2010)

7	**A Hybrid Model of Crime**	● Recurrent Neural Networks	● The hybrid model in this paper STARMA model has its advantages in predicting crime.

	Prediction	• Kernel Density Estimation The STARMA model, the hybrid model, has its benefits in predicting crime. On the one hand, the inclusion of LSTM in the model tackles the volatility of criminal data, making it more suitable for the STARMA model; on the other hand, it adds non-linear structure to the model and increases model fitting precision.This paper forecasts changes in the number of crimes each month and this approach enables us to make predictions by week, day, hour or even minute, in order to provide the public safety department with knowledge prediction reference in a timely manner	• This paper forecasts changes in the number of crimes each month and this approach helps them to make estimates by week, day, hour or even minute, in order to provide a timely guide to knowledge analysis for the department of public safety. • The model isn't flawless, and to develop the model it requires further research. • More effective algorithms should be explored to compare all possible order models and pick optimal parameters for the determination of space-time lag order in STARMA modeling. • More information on different locations is required to measure the weights more accurately
8	**Examining Deep Learning Architectures for Crime Classification and Prediction**	• Support Vector Machine • Mean Squared Error • Deep Learning To get a proper	• In this paper they examined DL methods' ability to predict hotspot areas in an urban setting, where crimes of some types are more likely to occur in a given future window. • To achieve this aim, we fed the DL methods

		assessment of the ability of the DL methods, we compare them with the state-of-the-art CCRBoost and ST- ResNet methods, as well as eight benchmark methodologies commonly found in the recent related literature.	with the minimum amount of data containing only details of the spatial, temporal and crime types. ● We used a dual output environment where the second output is the number of crimes that happened in the same future window in order for the models to accurately predict the order of "hotness" ● They selected our model configuration SFTT as the winning one compared to 10 different algorithms in 5 crime incidence datasets and further evaluated the parameters selected for robust results.
9	**Deep Learning for Real Time Crime Forecasting**	● Convolutional Neural Network ● Root Mean Squared Error ● Deep Neural Network In this job, we're applying the ST-ResNet on an hourly scale to real-time crime prediction. Due to the low regularity in space and time of the crime data, we perform data regularization both spatially and temporally. Our preprocessing of data and CNN approach increases the prediction accuracy significantly	The forecasting of real time crime is a major science and sociological problem. It has a strong impact on our quality of life. Mathematical simulation of crime has been the subject of recent efforts. Short et al developed novel differential equation models for the crime hotspots dynamics. ● Considering crime as self-exciting, Mohler et al applied to crime modeling the classical aftershock sequence of an outbreak type (ETAS) and its variants. These kinds of models give predictive power to a microscopic representation of the crime events. ● The models referred to above are built from historical data. There is also fascinating research on crime prediction using data from social networks, for example Twitter

		compared to applying the residual network on each individual grid. The change is due to the fact that the spatial information is no longer isolated in the convolutionary model allowing for the identification of the crime hotspot transitions. In space as well as time, our forecasts are highly accurate and can provide clear guidelines for crime prevention.	
10	**Dynamic Gesture Recognition Based on LSTM-CNN**	Batch Normalization (BN) is a regularization technique that aims to preserve the standard distribution of hidden layer activation values during training, which can accelerate network convergence The comparison experiment conducted consists mainly of comparing the efficiency of the LCNN, Convolution Neural Networks-Long Short Term Memory and the single long-short term	● The LSTMs model is used to derive knowledge about timing in the signals. ● The model of CNNs will perform a secondary extraction and classification of the signal features. ● Batch Normalization (BN) is a regularization technique that aims to preserve the standard distribution of hidden layer activation values during training, which can speed up network convergence. ● The authors design two experiments using the Myo Dataset and DB5 dataset obtained by themselves in NinaPro. The findings demonstrate the LCNN model's viability and efficiency benefit, by contrasting the accuracy of LCNN by standard methods on the basis of

		memory model based on Myo Dataset and obtaining the most appropriate deep learning model for gesture recognition based on quantitative indicators	the same dataset. ● Three models tend to be stable after 30 cycles but the convergence rate is different. LCNN has the fastest training time with little fluctuation, while CNN-LSTM has the slowest convergence rate with slight fluctuation throughout the process. ● The comparison experiment conducted is primarily to compare the efficiency of the LCNN, CNN- LSTM and the single MyoDataset-based LSTM model, and to obtain the most appropriate deep learning model for gesture recognition according to quantitative indicators.
11	**A Deep Learning Framework for Univariate Time Series Prediction Using Convolutional LSTM Stacked Autoencoders**	This paper presented a deep learning architecture for one-step time series prediction that integrates wavelet transformation, 2-dimensional convolutionary Long ShortTerm deep neural networks and stacked auto encoders The opposed time series is translated into a 2-Dimensional image sequence. This is done to make it appropriate for the identification of images by convolutional neural networks that are	● In this paper they usually find the estimation of the Time series as a difficult activity, which extends to several fields of endeavour. ● Analysts are concerned with forecasting stock market values, exchange rates, default loan rates or indexes on the stock market. ● In traffic control, one or more traffic parameters have to be anticipated to efficiently handle the traffic situation. ● A popular theme in the above-mentioned problems revolves around the need to evaluate a variable's past and/or current observations to estimate future observations values. ● Because of the broad significance and multidisciplinary application of time series,

		specialized in image recognition	there has been an increase in research studies aimed at developing techniques for predicting exact time series.
12	**Audio Replay SPoof Attack Detection using Segment- Based Hybrid Feature and Den-LSTM Network**	Products based on the automatic speaker verification system (ASV) have experienced exponential growth in recent years The automatic speaker verification system is vulnerable to spoofing attacks, especially the spoof replay attacks This is more difficult to detect audio replay spoof attack than the other attacks and it poses the greatest danger to the automatic speaker verification system.	The automatic speaker verification (ASV) device based products have experienced exponential development. ● The highest risk is the unauthorized access to spoofed speech. The ASV device is vulnerable to spoofing attacks, in particular spoof replay attacks. Replay spoof attack can be done without any specific technical experience or computer but just a cell phone. This is more difficult to detect audio replay spoof attacks than the other attacks and presents the greatest threat to the ASV program. Within this paper the authors concentrate mainly on detecting spoof attacks from audio replays
13	**Video Captioning With Attention Based LSTM and Semantic Consistency**	Graph Model element, MP-LSTM, S-VC, Soft-attention, S2VT, LSTME, p-RNN, MSVD dataset HRNE	An attention-based LSTM model with semantine consistency is proposed for converting videos to natural sentences. In particular, firstly, an attention mechanism is suggested that uses the dynamic weighted sum of local two-dimensional convolutional neural network representations. Then, to produce essential terms, an LSTM decoder takes these visual features at time t and the word embedding feature at time t−1. Finally, multimodal embedding is used to map the graphic and sentence features into a shared space to ensure the semantic

			continuity of the definition of the sentence and the graphic quality of the frame.
14	**Hierarchical LSTM with Adjusted Temporal Attention for Video Captioning**	Models such as pRNN, MP-LSTM, SA, HRNE, S2VT, GoogleNet, ResNet, Inception-v3 are evaluated on MSVD datasets and MSR- VTT.	Within this paper a hierarchical LSTM approach is proposed for video captioning with modified temporal focus (hLSTMat). The modified temporal focus is for determining whether to rely on knowledge about the visual information or the meaning of language. A hierarchical LSTMs is also designed to recognize both low-level visual information and high-level language background information at the same time to help the generation of video captions. The future research requires the implementation of the system to test the output of both temporal and visual features.
15	**Representation with Application to Captioning**	LSTM Embedding, RNN paragraph decoder on MSVD and M-VAD datasets	Input knowledge flow, and multiple consecutive inputs composite at a higher level. Then it uncovers temporal transitions with different granularities between frame parts, i.e. it can model the temporal transitions between frames, as well as the transitions between segments.
16	**Video Description using Bidirectional Recurrent Neural Networks**	Yao et al. In Microsoft Research Video Definition Corpus (MSVD) with scenes and BLSTM	Their solution was an encoder-decoder. They divided it into four stages, respectively using CNNs and LSTMs to identify images and to model their temporal relationship. Two items have been introduced at Encoder. Understanding which kind of objects and structures appear in images which is done by CNNs. Modeling over time their relationship and behavior. They proposed using a Bidirectional LSTMs layer to solve the second problem. Decoder consists of an LSTM network that acts as a model of language. They also used a method of concentrating attention in the

			appropriate objects.
17	**Video captioning with recurrent networks based on frame end video-level features and visual content classification**	They experimented with different models on the COCO dataset – cocok, coco-kf + cls, kf, kf+cls, traj, traj+cls and more. Kf – characteristics of the keyframe, complex course, cls- classification of material	Their research builds upon the method of static image captioning as suggested by Vinyals et. Al, and Karpathy et. introduced. Al. and extends this structure to images using both static image and video-specific functions. However, they concentrate on the utility of visual content classifiers as a source of additional information for creating captions. Learning generative models of phrases based on input image, video and class membership.
18	**A Simple Way to Initialize Recurrent Networks of Rectified Linear Units**	• Recurrent Neural Network • Mean Squared Error • Neural Network Recurrent neural networks (RNNs) are very effective dynamic systems and are the natural way to map an input sequence to an output sequence, as in speech recognition and machine translation, or to predict the next word in a sequence, as in language modeling A second purpose of this paper is to investigate whether rectified linear units can function well in recurrent neural networks and whether they can be easily modified in	• We performed TIMIT Phoneme analysis experiments with IRNNs and Bidirectional IRNNs and compared them with RNNs, LSTMs, and Bidirectional LSTMs and RNNs. Bidirectional LSTMs were previously applied to TIMIT in these experiments using the recorded recipe, we created phoneme alignments from Kaldi, and trained all RNNs with two and five hidden layers.Every model provided log Mel filter bank spectra with their delta and accelerations, where each frame was 120 (= 40 * 3) dimensional and trained to predict the state of the phone (1 of 180). From this task frame error rates (FER) are listed in table 4. • In this task we used 0.01I instead of initializing identity for the matrices of the IRNNs so we refer to them as iRNNs. Initiating with the full identity resulted in slow convergence, worse results and often led to divergence of the model during the training. We hypothesize that

		feedforward networks transferred to recurrent neural networks.	this was because related inputs to the neural net in adjacent frames are given in the speech task. Instead of paying attention mainly to the present data, the usual IRNN keeps adding this past information, since it has a difficult time forgetting the past. So not only do we show for the speech task that iRNNs function much better than RNNs composed of tanh units, but we also show that initialization with full identity is suboptimal if long-range effects are not needed. In such instances multiplying identity with a small scalar tends to be a good solution.

CHAPTER - 3

REQUIREMENT ANALYSIS & SOLUTION APPROACH

3.1 Overall Project Summary

This project aims at analyzing the crime pattern happening in a city / district and thereby predicting the crime hotspots for the city. Administrations, in fact, expend a lot of money by law enforcement departments seeking to avoid crimes occurring. Many law enforcement agencies today have vast amounts of crime-related data that are to be analyzed to convert into usable information[3].

Crime stats is complex as it has several dimensions and includes string records and narrative records in different formats, e.g. most. Despite this variety, using off-shelf, computational, and machine learning data analytics software is difficult to mine.This is the primary explanation why crime data mining lacks general forum. Though there are some propitiatory platforms for forecasting and analyzing crime data, they concentrate on very few non-extensible areas of crime, and do not include an API for integration with other tools[4].In addition, it is not possible to use the same method for the study and preparation, such as patrol beads and district borders.

We collected the crime data of CHICAGO, USA from the https://data.cityofchicago.org/. We selected crime data from the year 2012 to 2014 of all the crime types. The project then cleans the data collected as there are a lot of missing latitude and longitude using Google Geo-Spatial Finder. The useful data is then divided according to the number of crime events in a month. Crime hotspots are then created from the month wise crime data using ArcMAP software and later Photoshop is used to perfectly make the hotspots images. Convolutional Neural Network (CNN) is then applied to extract features from the hotspot files generated. The extracted features are then passed to Long Short Term Memory (LSTM). This algorithm helps the model to analyze and predict the crime hotspot for the next month. In this way, he could predict the crime hotspot for the next month using the crime hotspot of the previous month with the trained crime hotspot model.

The resulting shape file of the predicted crime hotspot is then compared with the actual crime hotspot image. Pixel by Pixel value is compared for both the images and the total difference for each pixel is aggregated to show the dissimilarity of the two images which helps us to know the percentage accuracy / loss for this model. This model can be integrated with any API for a city and police beats can be assigned to these crime hotspots to prevent the crime happening.

Also, the model developed was compared with the existing Machine learning models like Decision tree, SVM Classifier, KNN, Naive Bayes Classifier and PCA in terms of the Accuracy of the Crime Prediction. Same Crime dataset of Chicago was used which was used with the ConvLSTM model. All the Machine learning models use raw crime data. Loss and Accuracy Graph is being plotted for these models.

3.2 Requirement Analysis

The main requirements of the project can be listed in two categories: Non-Functional and Functional Requirements.

3.2.1 Non-Functional Requirements

Hardware Specifications:

- CPU: 2.80GHz Intel Core i7-7700HQ
- Graphics: NVIDIA GetForce GTX- 1050
- RAM: 16GB
- Storage: 4-GB DDR 5

Software Specifications:

- Operating System: Windows
- Programming Language Used: Python
- Tool: Google Collab & ArcGis
- Python Libraries:
  - import pandas
  - import numpy
  - import datetime
  - import os
  - import sys
  - import matplotlib.image
  - import tensorflow
  - from datetime import datetime

- from datetime import timedelta
- import matplotlib.pyplot
- import h5py
- import keras
- import gc
- from keras.optimizers import RMSprop,Adam
- from keras.layers import AveragePooling2D
- from keras.models import Sequential,load_model
- from keras.layers.convolutional import Conv3D
- from keras.layers.convolutional_recurrent import ConvLSTM2D
- from keras.models import Model
- from keras.layers import Input, Dense, MaxPooling2D, MaxPooling3D, Dropout, BatchNormalization, Flatten, Conv2D, Conv3D, AveragePooling3D, LSTM, Reshape
- from keras import backend
- from keras.callbacks import History
- import glob
- import cv2
- from sklearn.tree import DecisionTreeClassifier
- from sklearn.naive_bayes import GaussianNB
- from sklearn import linear_model
- from sklearn.metrics import mean_squared_error
- from sklearn.model_selection import cross_val_predict
- from sklearn.decomposition import PCA
- from sklearn.linear_model import LinearRegression
- from sklearn.model_selection import cross_val_score
- from sklearn import svm
- from sklearn.decomposition import PCA
- from sklearn.neighbors import KNeighborsClassifier
- from sklearn.metrics import f1_score
- from sklearn import linear_model

3.2.2 Functional Requirements

- Clean Data - The data which was extracted from the Chicago Crime Portal was not appropriate to be used for training. This is due to the fact that it contained many missing latitude/longitude and also some irrelevant features for each of the crime records.
- Annotated Data - The data has to be classified into month wise crime data. This month wise data is then used for making hotspots.
- All the libraries are to be pre-installed in the required Tool.
- Debugging of code to get rid of syntactical errors should be done.

3.3 Solution Approach

The model that is developed specifically targets the area of crime. This is capable of analyzing crime data in 3 different ways, namely Descriptive, Prescriptive, and Predictive Analytics. The most important element of this model is that, to help various forms of crime data analysis, it is built to be scalable. The model initially takes input requires a Crime dataset having a timestamp of the Crime Incident with Latitude & Longitude of the Crime Spot. The model will train itself according to the crime spot and will predict the future crime hotspots.

Initially, we collected the crime data of CHICAGO, USA from the https://data.cityofchicago.org/. This website is the official data portal of Chicago containing various data records of Buildings, Education, Environment, Events, Finance, Public Safety, Sanitation Parks & Recreation, Transportation etc. The Crime data file (2001 to present) received from the portal contains Date of the Crime Incident, Location of the incident, Beats, Ward, Block Address etc. but after understanding the data records, we found that there are a lot of missing Crime Location for a particular data record. We use Google API to find the missing Latitude/Longitude of the Crime Spots. As Google API provides 190 coordinates of the Address for an IP Address, hence we have to remove that missing entries from our Dataset.

We then selected Crime data records from the year 2012 to 2014 for our training process. The resultant data is then divided according to the number of crime events per month per. Separate CSV files are created from the original Dataset. Month wise Crime data CSV file is then passed to ArcMap software which is used for making various plots, Graphs, Pie charts, Hotspots etc. We use Kernel Density Crime hotspots tool present in the software to form crime hotspots monthwise. We use 0.001 cell size for hotspot creation. The crime hotspots are then masked with Chicago boundary and 5 classes of color range are used for Crime hotspot.

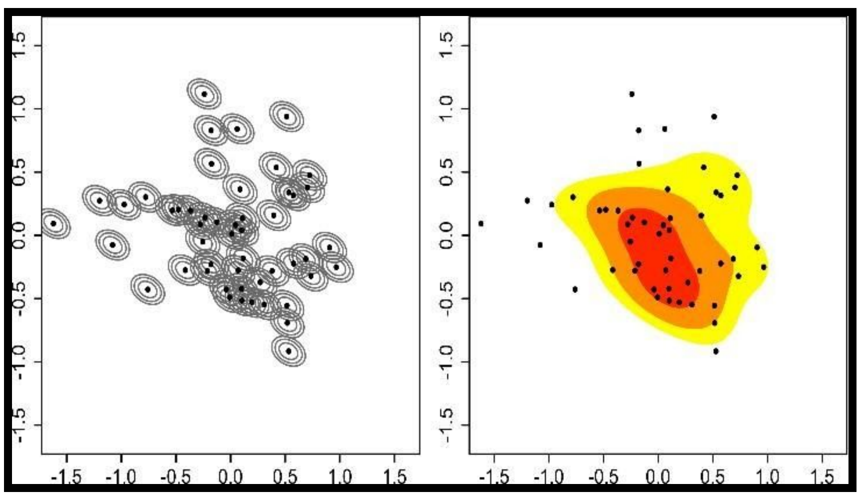


Figure 1. Spatio Kernel Density

The Crime hotspot images are then passed onto the Convolutional Neural Network(CNN) model. The objective of a Conv layer is to extract features of the input volume. The extracted feature is of the form of the original dimensions as we removed one of the phases of CNN which is Pooling. Pooling reduces our input image into a 1-d Array having features of the inputted images. The extracted feature is then passed onto the Long Short Term Memory (LSTM) model. Training is done by selecting the images in a group of 2 i.e. the first two images are trained and it will give a predicted crime hotspot image of the next month. In a similar way, We were able to predict the crime for the month of January,2015 using 36 months for training from 2012 to 2014. This algorithm helps the model to analyze and predict the crime hotspot for the next month. LSTM is the only model which considers long term dependencies. In this way, the model could predict the crime hotspot for the next month using the crime hotspot of the previous month with the trained crime hotspot model.

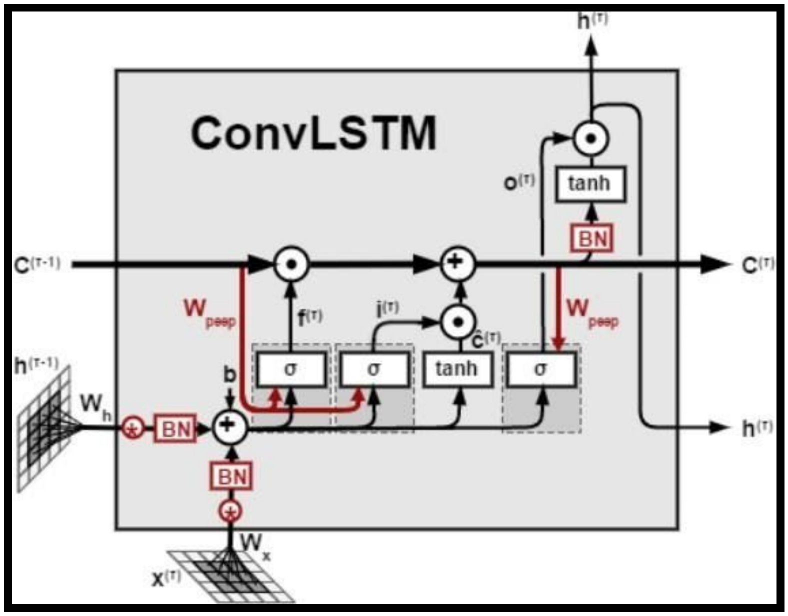


Figure 2. ConvLSTM Architecture

The resulting shape file of the predicted crime hotspot is then compared with the actual crime hotspot image. Pixel by Pixel value is compared for both the images and the total difference for each pixel is aggregated to show the dissimilarity of the two images which helps us to know the percentage accuracy / loss for this model. The accuracy model used here is the python inbuilt model which is used to find the similarity index between the two images using OpenCV. This model can be integrated with any API for a city and police beats can be assigned to these crime hotspots to prevent the crime happening.

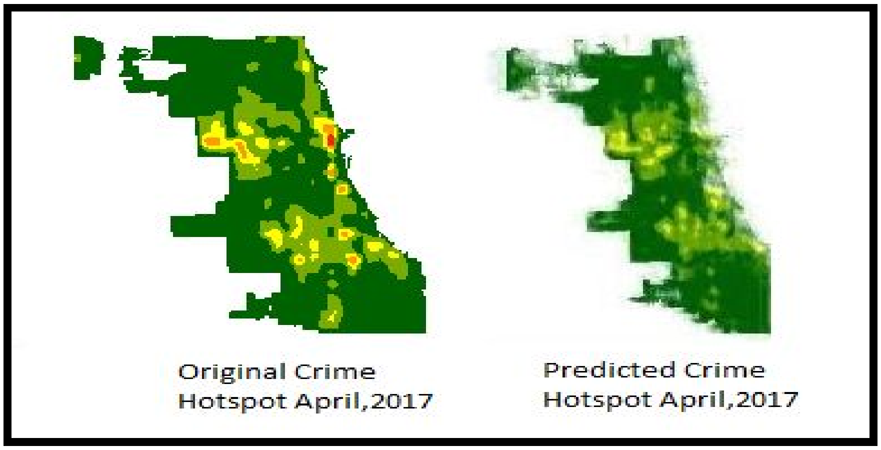


Figure 3. Original vs Predicted Hotspot Image

Later, to prove whether our model is working accurately and giving better results, we compared our model with the existing models like SVM classifier, Decision tree, Naive Bayes Classifier, PCA and KNN. For each of the machine learning models, we compare the accuracy of the crime predicted by the model. All these models take raw crime data as input and we imputed the same crime dataset which we use for our model for proper comparison. Loss and accuracy graphs are being plotted for these models. Hence after comparing the results, it can be said that our model shows a significant increase in the rate of the accuracy of the prediction.

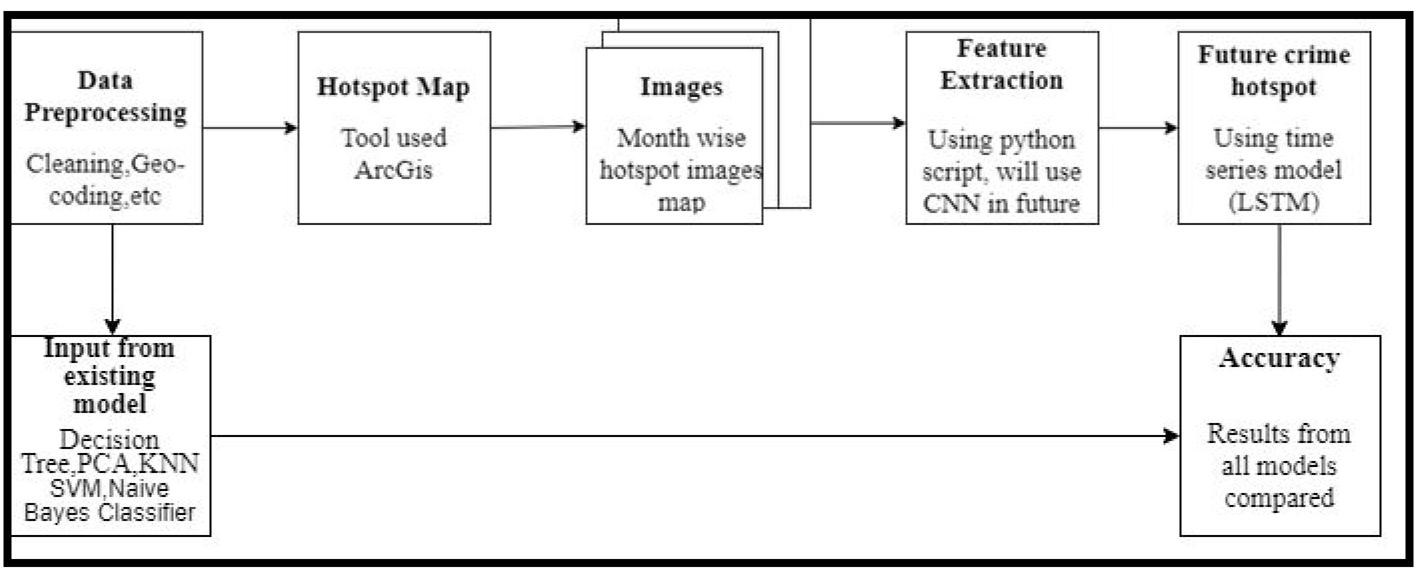


Figure 4. Sequence Diagram

CHAPTER - 4

PROJECT MODELLING & IMPLEMENTATION DETAIL

4.1 Design Diagrams

4.1.1 User Case Diagrams

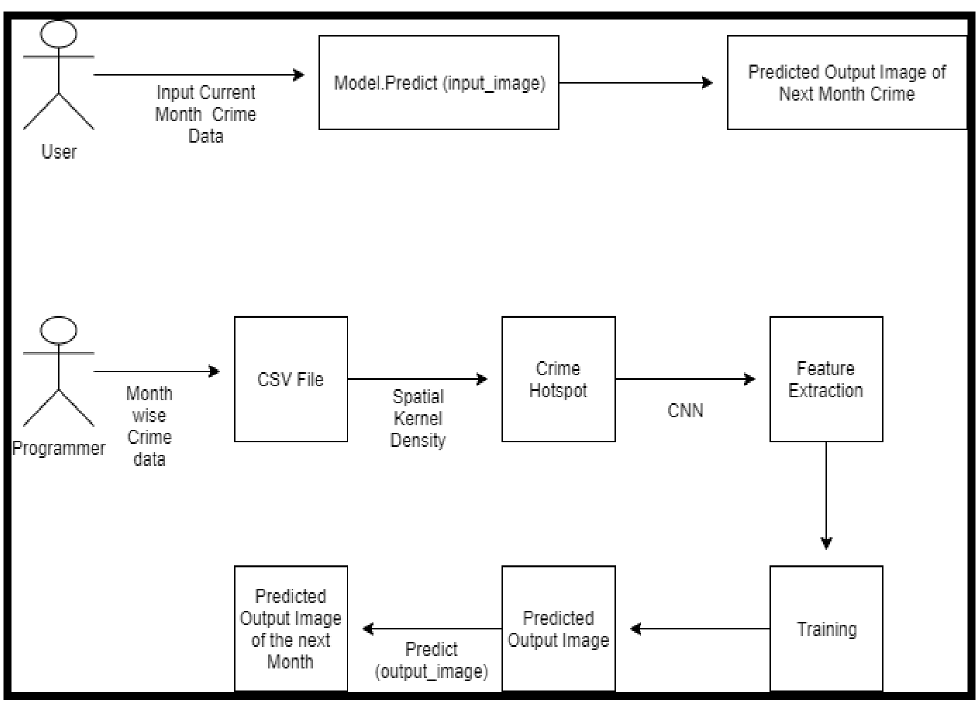


Figure 5. User Case Diagram

4.1.2 Control Flow

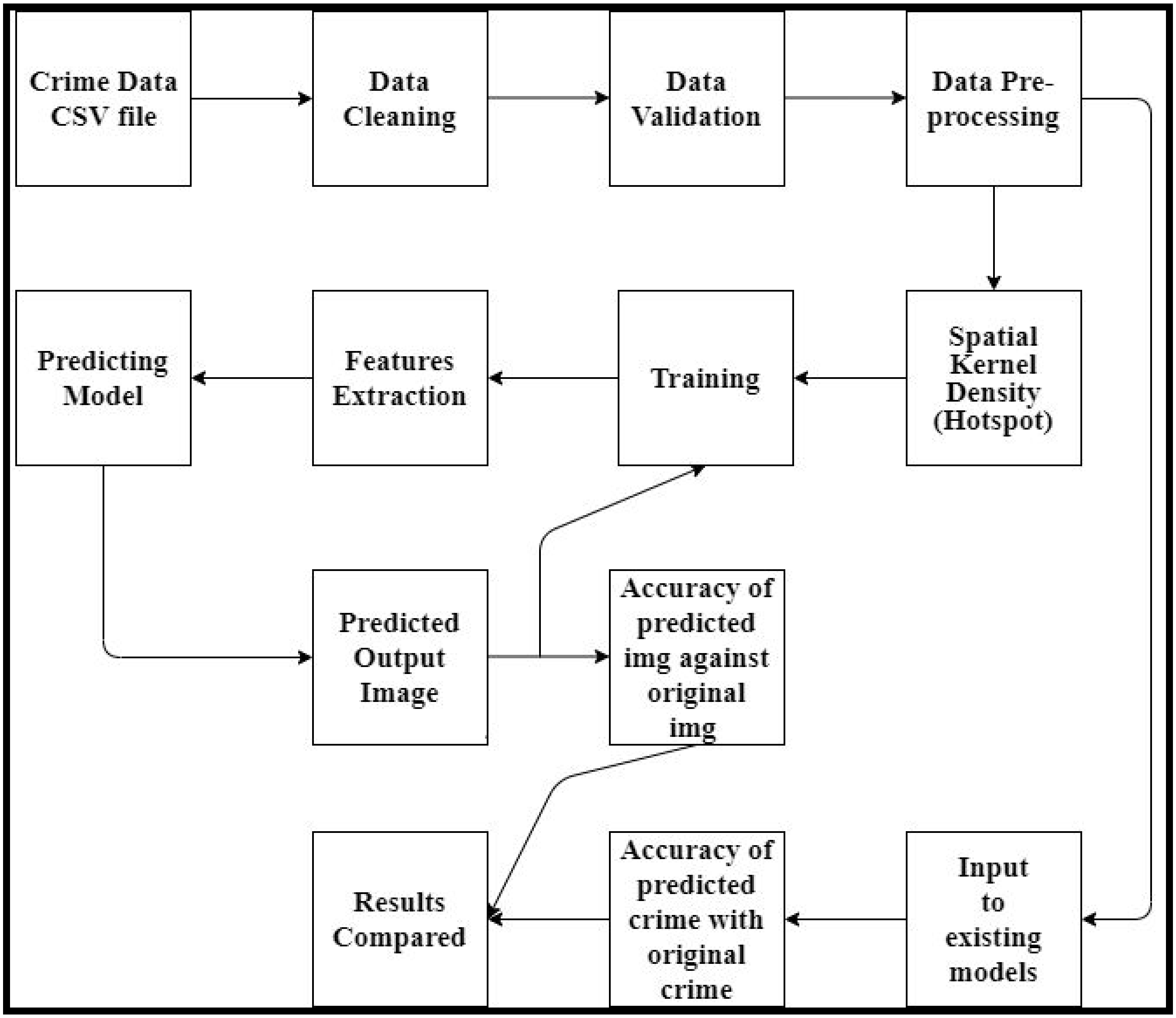


Figure 6. CONTROL FLOW DIAGRAM

4.2 Implementation Details

4.2.1 Overview

Significant quantities of data are gathered regularly by many law enforcement agencies around the world. It can be really interesting to examine this data and see what sort of patterns they can make. Using Machine Learning (ML) can use this data for several predictive tasks.Through identifying the most vulnerable targets to be targeted over a fixed period of time and geographic area, police would be able to identify effective ways to utilize the available resources, as well as discovering and resolving the issues that contribute to crimes.And to find a way to minimize crime, we look at ways to use ML to predict the types / categories of crime that are most likely to occur within a given time and place.

4.2.2 Dataset

a) About Dataset

From 2001 to present, we picked a Chicago, USA crime data to demonstrate prediction of the crime category. The data set has been downloaded from https:/data.cityofchicago.org/ Open data portal[34] and consists of nine attributes:

1. Date - timestamp of the crime incident (Time Stamp)

2. Block – Description of the crime spot (String)

3. Category - category of the crime incident (String)

4. District – District of Police Department which the incident occurred (String)

6. Beat – Police District Car Number (String)

7. Ward – Nearest Police Ward (String)

8. Longitude (double)

9. Latitude (double)

10. ID - Registered Crime Event Case Number

b) Pre-processing & Cleaning

The data was extracted from the Official Data Portal of Chicago where Data regarding buildings, Education, Environment, Events, Finance, Public Safety, Sanitation Parks & Recreation, Transportation are present. We selected Crime Data from 2001 to present. The Data File has more than 60 lakhs crime records. The data File (.csv) is downloaded and used for the project. The file contains a lot of missing Latitude/Longitude of the crime records. Some of the missing data entries are found using Google API but the constraint was that The software could only find 180 missing entries for an IP Address. The Updated file is then divided into month wise crime data records. The separated Month wise data file is then used for creating Crime hotspots.

4.2.3 Methodology

c) Feature Selection

Feature Selection is the initial step where we consequently or physically select those features which contribute most to our predicted variable. Having superfluous features in the data can diminish the exactness of the models and cause the models to learn dependent on unessential features. Therefore, it is necessary to select only those features which are most accountable for the prediction. For the model, we selected the location of the crime incident only keeping all the other parameters like Police Stations, Bars, Vacant Spaces etc. constant.

d) Data Visualization

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data. We mapped Month wise crime data and created hotspots of the crime events using ArcMap. We applied Kernel Density Hotspot Estimation for creating the crime hotspots with 0.001 cell size i.e. block size for each cell in an image. We selected 5 Classes for the color range for our hotspot. We then mask the crime hotspot with the boundary of Chicago.

e) Feature Extraction

Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing. A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process. Extracted features can be grouped in various ways, such as time or frequency-domain features, linear or non-linear features, unimodal or multimodal features, etc. Considering computational complexity, extracted features range from simple statistical features (e.g. mean, standard deviation) to often modality-dependent complex features. For our model, Convolutional Neural Network (CNN) is used for feature extraction from the crime hotspots. CNN has different phases. It has two main parts: feature learning (Conv, Relu,and Pool) and classification (FC and softmax).

d) Training

Now we have acquired the data which can be trained and used for crime analysis and prediction. The hotspot data is then uploaded to Google Drive and data is then taken from Google Drive directly. We resize the original hotspot images (size 1073*702) into size of 200*200. All other variations are being generalized in between 0 to 255. The data is then passed onto the ConvLSTM model. We created a batch of size 3 with time frame of 2 and kernel size of 3*3 for the training. We consider different filters at different training steps. We also use Adam optimizer with a learning rate of 0.01 and 0.1 and Mean Square Error as loss function for the model. For the total batch, we run the complete models for 1000times and train our Crime Model. Model is saved at every 2 iterations of the total batch trained and .pynb_checkpoint file is saved for the Model.

4.2.4 Comparison with the Existing Model

The model developed is compared with the existing machine learning models like SVM, PCA, Decision tree, KNN, Naive Bayes Classifier. Same dataset was sent as input to these models and results are then noted. The accuracy of these models are compared to prove that our model outperforms all other models in terms of accuracy, training and time of prediction. Also, the Loss and Accuracy curve of these models were plotted to compare the accuracy with our model.

CHAPTER -5

TESTING

5.1 Testing Plan

There are few types of classification which have been applied to test the model. Example we have trained a model using the crime hotspot images which is divided into batch sizes of 3. Now, we test the trained model with the help of the data of the previous month found using the training model. The predicted image is passed as an input for the next month and passed into the batch and then results are then found. For the next month prediction, you have to input the image of the previous month and the model will predict the crime hotspot using the previous trained data and the new inputted image This is equivalent to real time testing as we have trained a model using attributes gathered from crime images and then predict that the new hotspot for next month and used the trained model to test it on the next month image.

A ConvLSTM network with 4 Conv2D layers and 4 ConvLSTM2D layers are used for training and predicting the new crime hotspots.

Here we can train the data using 3 year of crime data i.e. 36 months and then testing crime for the next month using the trained model and previous month output. For batch1 of size 3, Training is done for the first 2 months and then crime is predicted for the 3rd month. For batch2, training is done on the 2nd and the predicted 3rd month and we will find the predicted crime hotspots for the 4th month. Similarly, the training is done for 3years and the last batch will predict the crime hotspot for 4year and 1st month. For testing the last predicted image is passed as a new batch and the model will predict the result for its next month.

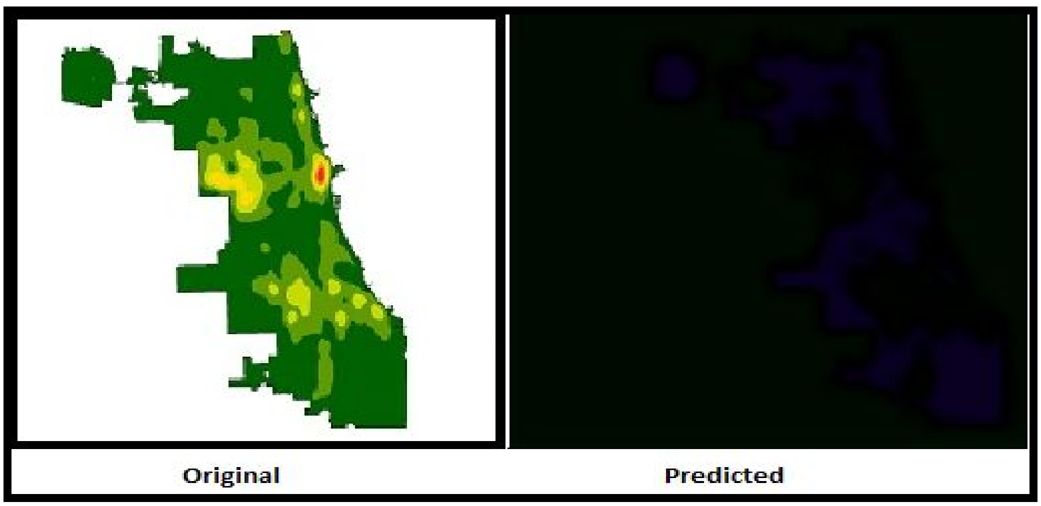
5.1.1 Component decomposition and type of testing required

Type of Test	Will Test Be Performed?	Comments/Explanations	Software Component
Unit	YES	Specify and test one point of the contract of a single method of a class.	The unit testing was performed on the functions of the code.
Performance	YES	Is the process of determining the speed, responsiveness and stability of a computer.	The results improved as tested with different small kinds of dataset.
Stress	YES	verified the stability & reliability of the system	The model is trained with thousands of iterations and now the trained model is a bit stable but not completely.
Security	YES	uncovers vulnerabilities of the system	The exploration rate is high. Thus, vulnerabilities are well handled.

5.2 List all test cases in stipulated format

The training is being done by changing the number of the Convolutional layer and ConvLSTM. Following are the changes and the corresponding output image of the selected Training Parameters:

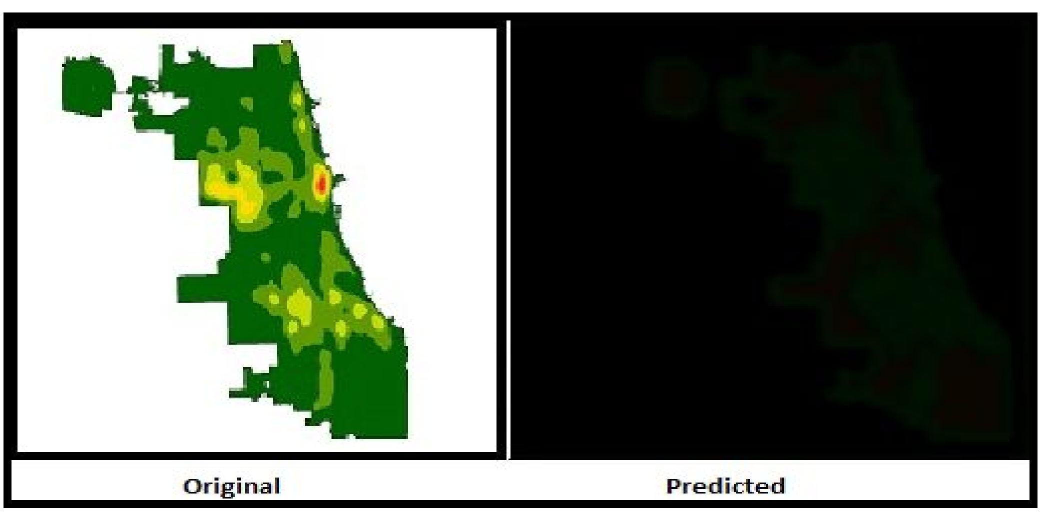
- **3 Conv2D and 3 ConvLSTM2D**



The predicted Image when compared with the actual image shows only 78 percent similarity.

Figure 7. Test Case 1

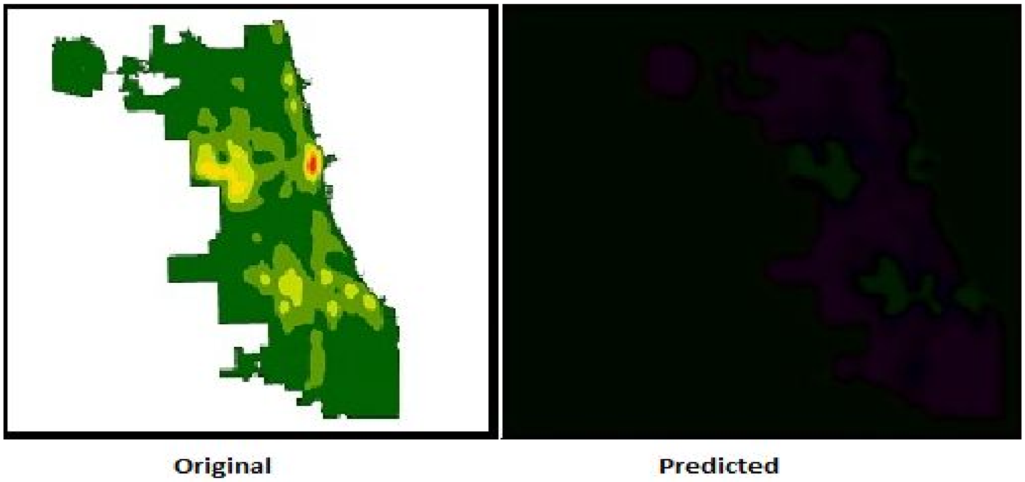
- **4 Conv2D and 3 ConvLSTM2D**



The predicted Image when compared with the actual image shows only 81 percent similarity.

Figure 8. Test Case 2

- **4 Conv2D and 4 ConvLSTM2D**



The predicted Image when compared with the actual image shows only 83.75 percent similarity.

Figure 9. Test Case 3

------> Also, there is a limitation in the Keras API which was that the color integration of keras is very poor. As blue color has the least wavelength among the three most pronounced colors which are red, yellow and blue. So, the color distortion happened while training the images which causes its upper and background color to get distorted. Hence we were not able to get the perfect hotspot image as we assumed.

5.3 Error and Exception Handling

CASE 1: Resize the image to 600*600

The model is unable to run and all the resources assigned by the Google Collab gets lost. This results in some of the features to get lost.

CASE 2: Passing data which is not preprocessed

The data which is not clean cannot be classified by the classifier properly.

CASE 3: Shutting of the Google Collab

If the Google Collab gets closed then all the processing gets lost.

CASE 4: Unable to connect to Google Drive

Sometimes authentication takes too long to get connected to Google Drive. Hence the process takes a lot of time and resources gets over and crashes.

Case 5: Less RAM available

Google Collab initially provides 7GB of Ram. Hence processing of images becomes very difficult and even training crashes in between.

5.4 Limitation of the Tool / Solutions

This approach is highly speculative, and challenges decades-worth of established research for predicting crime for a city. However, the solution is not fully accurate as we do not consider environmental factors which affect crime incidents. Also, due to less RAM available, the training process took a lot of time. Google Collab requires Internet facility to be available at all times. Hence checkpoints need to be saved properly for future execution of the project.

The hotspot created from ArcMap sometimes are not masked with boundaries properly. Hence we have to remove them as it will affect the training and output of the model. Also, some of the entries are missing latitude and longitude which results in training deviation. We use Google Location FInder to find the missing crime entries location but due to App limitation of finding max. of 190 entries per IP address causes the majority of missing entries to be deleted from the Dataset chosen.

Also, there is a limitation in the Keras API which was that the color integration of keras is very poor. As blue color has the least wavelength among the three most pronounced colors which are red, yellow and blue. So, the color distortion happened while training the images which causes its upper and background color to get distorted. Hence we were not able to get the perfect hotspot image as we assumed.

CHAPTER - 6

FINDINGS, CONCLUSION & FUTURE WORKS

6.1 Findings

Implementation of the CNN and LSTM was in such a way that approximately 90 percent accuracy was found. The results are much better when compared to the results of a single approach or when these algorithms are used with some other models. Considering the fact that we do not consider the environmental factors which play an important role in crime prediction, we still achieved better results and performance is also better than existing models. There had been an improvement in performance of CNN a few months back which was not considered in this model. The project helped us in understanding the basic and advanced level concepts used in Deep Learning and its applications. Accuracy can further be improved if the training process takes a bit less time. Also, the model is compared with existing models and our model outperforms all previous models in terms of Accuracy, Time of prediction and Training.

● Accuracy and Comparison with Existing Models

1. Decision Tree Classifier

Figure10. Decision Tree Classifier

In the Decision Tree Classifier, we can get the best number of max depth feeding according to the plot above. The mean of cross val score accuracy starts to rise with the growing number of max depths and then begins to decrease on number 3.In DecisionTreeClassifier, max depth=3 will therefore get the best results in analyzing the dataset. Hence we can only select max of 3 different types of crime which will give best results. Considering all types of crimes, it tooks 7 days to train itself to half of its levels. ALthough the accuracy is near 83% which is approximate to average accuracy in our model.

2. Naive Bayes Classifier

```
In [124]: from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB()
y_pred = gnb.fit(initial, Y).predict(initial)

print("mislabel num is ",(Y != y_pred).sum())

# print ('sigma is ',gnb.sigma_)
variance=gnb.sigma_
stand_deviation =np.sqrt( variance)
# print('standard deviation is',stand_deviation);
sum_standard=stand_deviation[0]+stand_deviation[1]
# print('sum of standard deviation is',sum_standard);

# print ('theta is ',gnb.theta_)
mean=gnb.theta_
difference=mean[0]-mean[1]
# print('difference is ',abs(difference))
normalized_feature=abs(difference)/sum_standard
# print('normalized_feature is ',normalized_feature)


ind = np.argpartition(normalized_feature, -10)[-10:]

print('ind is ',ind)
print('10_max_normalized_feature is ',normalized_feature[ind])

for x in range(0, len(ind)):
    index=ind[x]
    print(index)
    print('feature_name[index] is ',feature_name[index])




from sklearn.model_selection import cross_val_score
fold=df['fold']
scores = cross_val_score(gnb, initial, Y,fold,'accuracy',10)
print('cross_val_accuracy is ',scores)
print('cross_val_accuracy_avg is ',np.array(scores).mean())
scores = cross_val_score(gnb, initial, Y,fold,'precision',10)
print('cross_val_precision is ',scores)
print('cross_val_precision_avg is ',np.array(scores).mean())
scores = cross_val_score(gnb, initial, Y,fold,'recall',10)
print('cross_val_recall is ',scores)
print('cross_val_recall_avg is ',np.array(scores).mean())

mislabel num is  442
ind is  [38 44 45 41 15 46 50  3 43 40]
10_max_normalized_feature is  [ 0.61686366  0.80974842  0.66500857  0.67464461  0.66107643  0.64294945
  0.70926105  0.73522995  0.74554481  0.69397809]
38
feature_name[index] is  MalePctDivorce
44
feature_name[index] is  PctKids2Par
45
feature_name[index] is  PctYoungKids2Par
41
feature_name[index] is  TotalPctDiv
15
feature_name[index] is  pctWInvInc
46
feature_name[index] is  PctTeen2Par
50
feature_name[index] is  PctIlleg
3
feature_name[index] is  racePctWhite
43
feature_name[index] is  PctFam2Par
40
feature_name[index] is  FemalePctDiv
cross_val_accuracy is  [ 0.775       0.8         0.825       0.79899497  0.70351759  0.65326633
  0.81407035  0.73366834  0.71356784  0.79899497]
cross_val_accuracy_avg is  0.761608840201
cross_val_precision is  [ 0.86363636  0.92929293  0.95        0.92079208  0.94594595  0.86842105
  0.92307692  1.          0.77868852  0.93814433]
cross_val_precision_avg is  0.911799814828
cross_val_recall is  [ 0.76   0.736  0.76   0.744  0.56   0.528  0.768  0.576  0.76   0.728]
cross_val_recall_avg is  0.692
```

Figure 11. Naive Bayes Classifier

On average Naive Bayes has poorer accuracy and recall on the dataset than Decision Tree. This is possibly due to associations between features in the dataset that are conditionally independent, as presumed by the Naive Bayes classifier.   Interestingly, both agree that AvgCrimeprRegion is the most predictive function.

3. Support Vector Classifier

```
In [125]: from sklearn import svm
          lin_svc = svm.LinearSVC(C=0.01447, penalty="l1", dual=False).fit(initial, Y)
          # using L1-norm (sparsity method) to make unless feature weight become 0 , C value increase->more complex
          model having more weight
          feature_weight=abs(lin_svc.coef_[0])
          print("",feature_weight)
          for i in range(0,len(feature_weight)):
              if(feature_weight[i]!=0):
                  print('select_feature_is ',feature_name[i], ' feature_weight is ', feature_weight[i])



          from sklearn.model_selection import cross_val_score
          fold=df['fold']
          scores = cross_val_score(lin_svc, initial, Y,fold,'accuracy',10)
          print('cross_val_accuracy is ',scores)
          print('cross_val_accuracy_avg is ',np.array(scores).mean())
          scores = cross_val_score(lin_svc, initial, Y,fold,'precision',10)
          print('cross_val_precision is ',scores)
          print('cross_val_precision_avg is ',np.array(scores).mean())
          scores = cross_val_score(lin_svc, initial, Y,fold,'recall',10)
          print('cross_val_recall is ',scores)
          print('cross_val_recall_avg is ',np.array(scores).mean())
```

```
 [ 0.          0.          0.14001883  0.60158853  0.          0.29846094
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.
  0.20685957  1.40999677  0.          0.          0.5248893   0.          0.
  0.          0.          0.          0.37405216  0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.18042693  0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.17249698  0.          0.
  0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.        ]
select_feature_is  racepctblack  feature_weight is  0.140018826615
select_feature_is  racePctWhite  feature_weight is  0.601588527123
select_feature_is  racePctHisp  feature_weight is  0.298460944212
select_feature_is  FemalePctDiv  feature_weight is  0.206859566967
select_feature_is  TotalPctDiv  feature_weight is  1.40999677397
select_feature_is  PctKids2Par  feature_weight is  0.524889297296
select_feature_is  PctIlleg  feature_weight is  0.374052163259
select_feature_is  PctPersDenseHous  feature_weight is  0.180426932386
select_feature_is  MedRentPctHousInc  feature_weight is  0.172496983106
cross_val_accuracy is  [ 0.775       0.875       0.87        0.85427136  0.73366834  0.69849246
  0.77386935  0.85427136  0.83919598  0.79899497]
cross_val_accuracy_avg is  0.80727638191
cross_val_precision is  [ 0.75641026  0.84615385  0.88372093  0.88188976  0.90909091  0.81553398
  0.78169014  0.96153846  0.82068966  0.81021898]
cross_val_precision_avg is  0.846693692191
cross_val_recall is  [ 0.944  0.968  0.912  0.896  0.64   0.68   0.88   0.816  0.952  0.888]
cross_val_recall_avg is  0.8576
```

Figure 12. Support vector Classifier

For this dataset, SVC increases precision, accuracy and recall compared to Decision Tree. This may be explained by the fact that SVC can find an ideal linear separating hyperplane(Crime type and Avgcrimeprlocation) for the dataset while Decision Tree can only use axis-aligned planes to splint the data set in a hierarchical manner (Crime types).

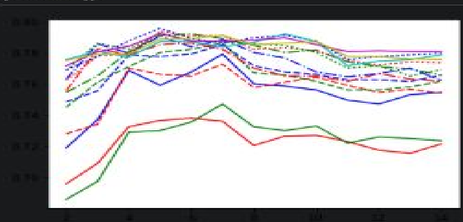
4. PCA & K-NN

```
In [133]: from sklearn.decomposition import PCA
          from sklearn.neighbors import KNeighborsClassifier

          df=pd.read_csv('communities-crime-clean.csv')
          df['highCrime'] = np.where(df['ViolentCrimesPerPop']>0.1, 1, 0)
          initial=pd.read_csv('communities-crime-clean.csv')
          Y = df['highCrime']
          fold=df['fold']
          state=df['state']
          community=df['communityname']
          initial = initial.drop('fold', 1)
          initial = initial.drop('state', 1)
          initial = initial.drop('communityname', 1)
          initial = initial.drop('ViolentCrimesPerPop', 1)

          x=[]
          y=[]
          for k in range (1,16):
              y.append([])
          for i in range (2,15):
              pca = PCA(n_components=i)
              pca.fit(initial)
              pcdf = pca.transform(initial)
              for j in range (1,16):
                  knn = KNeighborsClassifier(j)
                  knn.fit(pcdf,Y)
                  scores = cross_val_score(knn,pcdf,Y,fold,'accuracy',10)
                  y[j-1].append(np.mean(scores))
              x.append(i)

          plt.plot(x,y[0],'r-',
                   x,y[1],'g-',
                   x,y[2],'b-',
                   x,y[3],'r--',
                   x,y[4],'g--',
                   x,y[5],'b--',
                   x,y[6],'r-.',
                   x,y[7],'g-.',
                   x,y[8],'b-.',
                   x,y[9],'r:',
                   x,y[10],'g:',
                   x,y[11],'b:',
                   x,y[12],'c-',
                   x,y[13],'m-',
                   x,y[14],'y-')
          plt.show()
```



- X-axis is number of components
- Y-axis is accuracy
  - Red-solid-line is k=1
  - Green-solid-line is k=2
  - Blue-solid-line is k=3
  - Red-dashed-line is k=4
  - Green-dashed-line is k=5
  - Blue-dashed-line is k=6
  - Red-dash-dot-line is k=7
  - Green-dash-dot-line is k=8
  - Blue-dash-dot-line is k=9
  - Red-dotted-line is k=10
  - Green-dotted-line is k=11
  - Blue-dotted-line is k=12
  - Cyan-solid-line is k=13
  - Magenta-solid-line is k=14
  - Yellow-solid-line is k=15

Figure 13. PCA & K-NN

The findings for PCA to K-NN are identical to those for Decision Tree. The most important features are significantly different from the other methods implemented in this project, perhaps because they are not derived from a classification algorithm but from the features with the greatest variance in an algorithm for dimension reduction. The accuracy of this model is almost 81.44%.

- **Average Loss per epochs in training and Validation Process**

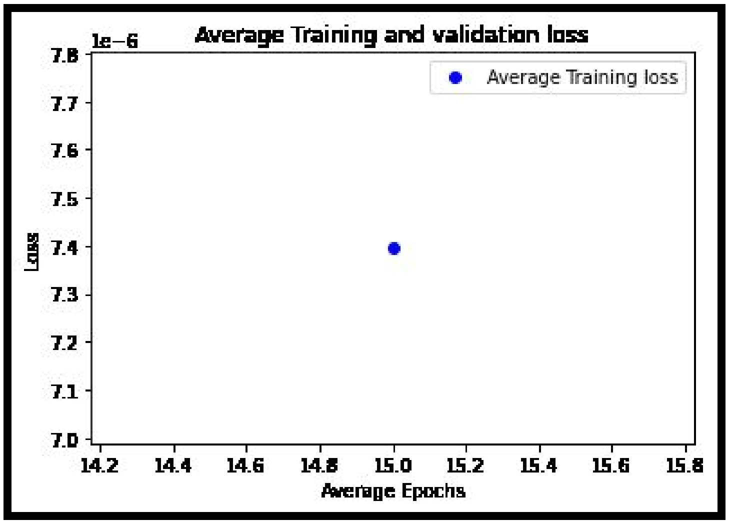


Figure 14. Average Training & Validation Loss

- **Training results**

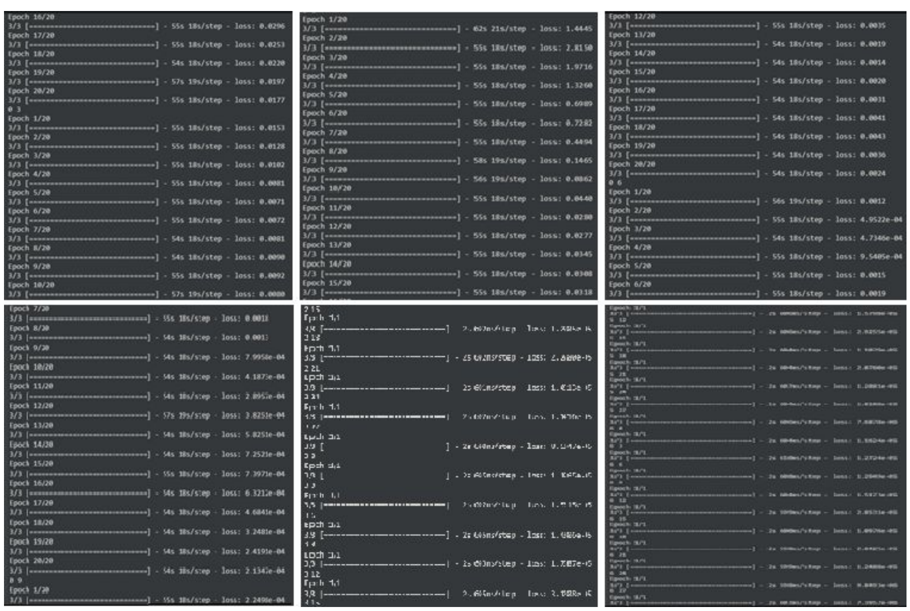


Figure 15. Training Results

- **Accuracy Table**

S.No.	Model Used	Avg. Accuracy
1.	**ConvLSTM**	**81.75%**
2.	**PCA & K-NN**	**78%**
3.	**Support Vector Classifier**	**80%**
4.	**Decision Tree Classifier**	**78%**
5.	**Naive Bayes Classifier**	**77%**

6.2 Conclusion

As shown in the review of related works several attempts have and are being made to predict crime in order to make crime prevention more efficient. In some areas, the prediction methods have also been implemented in the daily police work with successful results. Although the results are promising, there is still enough room for further improvement. This work is a first step into a potential future where the Chicago Police authority utilizes modern and innovative solutions to help reduce crime rates and prevent crime happenings. It is also a lookout to new technologies where machine learning is becoming more understood, sophisticated and used to greater extent. With more refined data processing, storage and analysis methods comes new possibilities to utilize the data that the police had gathered for years. This is a natural step in today's digitization efforts.

One potential usage of the predicted risk areas is to send police to patrol the area if there is nothing more important to attend. However, limited resources will probably make this a neglected task. Instead, this could be used in an ecosystem of crime preventive parties where municipalities, neighborhood watches and individual citizens cooperate with the police to reduce crime rates in their areas.

Regardless of insufficient resources, we still managed to apply the algorithm selected to the most. Considering the fact that we do not consider the environmental factors, our model and algorithm performs better both in terms of Complexity and Performance than the existing models. The accuracy of our model was found out to be around 90 percent while maximum accuracy of other models were found to be 75-85 percent with the same dataset. Also, due to shortage of data and many missing entries, we have to remove those data records from our crime dataset.

During the course, we explore various state-of-the-art algorithms for crime analysis and prediction, their advantages as well as their drawbacks. From this we can conclude that most of these algorithms use supervised and reinforcement learning with CNN and RNN networks. For feature extraction, CNN can be used along with Grid based Prediction and other techniques. RNN is used for predicting time series models but it doesn’t account for long term dependencies due to vanishing and exploding gradients. But nowadays crimes are dependent on various Environmental and Point of Interest anomalies in the city. Introduction of algorithms like LSTM, GRU shows significant increase in performance and are capable for long term dependencies. After a deep research and implementation, it can be concluded that our model and algorithm performs better both in terms of Complexity and Performance. The accuracy of our model was found out to be around 90 percent.

6.3 Future Work

This is a first prototype of a prediction model which could be implemented to predict and furthermore prevent crime. However, some things are left unfinished or unperformed due to lack of time and resources. In the design of the architecture for the prediction model there are a wide range of parameters that needs to be configured. These parameters have been identified and selected based on related works studies, empirical tests and best practice information. Further work into the design of the architecture and configuration of the model's parameters may improve the accuracy of the predictions. Considering Environmental factors such as Police Stations, Bars, School, Parks, Recreational Buildings etc. as constant can completely change the output of crime prediction models.

By using the uniform grid instead of administrative regions the data is aggregated to areas disregarding natural dividers such as water, larger roads or forests. For the convolutional operations to work the grid structure with eight neighbors per cell is needed. However, there could be potential in using administrative regions and defining each region's neighbors using a graph structure with the condition of eight neighbors. However, creating a division like this compatible with convolutional operations is not an easy task.

To implement the model in daily police work, a seamless implementation needs to be configured. We propose a web map service which trains during the night and delivers new risk areas for the current day in the morning. Automatic scripts for collecting, processing, training and visualization will need to be implemented to deliver an easily interpreted risk area map for normal police officers and managers. Studies on the density requirements of areas need to be performed, as rural areas with very few amount of burglaries will be difficult to predict.

REFERENCES

[1]. Timothy Hart, Paul Zandbergen. Kernel Density estimation and hotspot mapping. (PIJPSM 37,2).

[2]. Yujie Hu, Fahui Wang, Cecile Guin, Haojie Zhu. A spatio-temporal kernel density estimation framework for predictive crime hotspot mapping and evaluation. Y. Hu et al. ,2018.

[3]. Nurul Hazwani Mohd Shamsuddin, Nor Azizah Ali, Razana Alwee. An Overview on Crime Prediction Method. 2017 IEEE.

[4]. Lian Duan,Tao Hu, En Cheng, Jianfeng Zhu, Chao Gao. Deep Convolutional Neural Networks for Spatiotemporal Crime Prediction. Int'l Conf. Information and Knowledge Engineering.

[5]. Ke Wang, Peidong Zhu, Haoyang Zhu, Pengshuai Cui, and Zhenyu Zhang. An Interweaved Time Series Locally Connected Recurrent Neural Network Model on Crime Forecasting. K. Wang et al., 2017.

[6]. Murtaza Roondiwala, Harshal Patel, Shraddha Varma. Predicting Stock Prices Using LSTM. (IJSR 2017).

[7]. Rui Fu, Zuo Zhang, Li Li. Using LSTM and GRU Neural Network Methods for Traffic Flow Prediction. 31st Youth Academic Annual Conference of Chinese Association of Automation.

[8]. Maja Kalinic, Jukka M. Krisp. Kernel Density Estimation (KDE) vs. Hot-Spot Analysis - Detecting Criminal Hot Spots in the City of San Francisco. AGILE 2018 – Lund.

[9]. Mohammad A. Tayebi, Martin Ester, Patricia L. Brantingham, Uwe Glasser. Spatially Embedded Co-Offence Prediction Using Supervised Learning. (IKE' 17).

[10]. Suhong Kim, Param Joshi, Parminder Singh Kalsi, and Pooya Taheri. Crime Analysis Through Machine Learning. 2018 IEEE.

[11]. Shyam Varan Nath. Crime Pattern Detection Using Data Mining. arXiv preprint arXiv:1508.00941, 2015.

[12]. Lian Duan,Tao Hu, En Cheng, Jianfeng Zhu, Chao Gao. Deep Convolutional Neural Networks for Spatiotemporal Crime Prediction. Int'l Conf. Information and Knowledge Engineering.

[13]. Ying-Lung Lin, Meng-Feng Yen, Liang-Chih Yu. Grid-Based Crime Prediction Using Geographical Features. International Journal of Geo- Information.

[14]. Yong Zhuang, Matthew Almeida, Melissa Morabito, Wei Din. CrimeHotSpotForecasting: A Recurrent Model with Spatial and Temporal Information. 2017 IEEE International Conference on Big Knowledge.

[15]. Bao Wang, Duo Zhang, Duanhao Zhang, P.Jeffrey Brantingham, Andrea L. Bertozzi. Deep Learning for Real Time Crime Forecasting. (IKE' 15).

[16]. Alexander Stec, Diego Klabjan. Forecasting Crime with Deep Learning. arXiv:1806.01486v [stat.ML] 5 Jun 2018

[17]. ChaoHuang, JunboZhang, YuZheng, NiteshV.Chawla. Deep Crime- Attentive Hierarchical Recurrent Networks for Crime Prediction. IEEE 2018.

[18]. Panagiotis Stalidis, Theodoros Semertzidis, Member, IEEE and Petros Daras, Senior Member, IEEE. Examining Deep Learning Architectures for Crime Classification and Prediction.

[19]. Meilin Liu, Tianliang Lu. A Hybrid Model of Crime Prediction. IOP Conf. Series: Journal of Physics: Conf.

Series 1168 (2019) 03203.

[20]. Sukanchalika Boontham, Phayung Meesad. Time Series Analysis of Stock Prices based on Deep Learning.

IEEE 2016.

[21] P. Wang, R. Mathieu and H.J. Cai, "Predicting Criminal Recidivism With Support Vector Machine," in International Conference on In management and Service Science, 2010, pp. 1-9.

[22] Q. Li-Na, "Software reliability prediction model based on PSO and SVM," in International Conference on Consumer Electronics, Communications and Networks, 2011, pp. 5236-5239.

[23] K. Kianmehr and R. Alhajj, "Crime Hot-spots prediction using support vector machine," in IEEE International Conference on Computer Systems and Applications, 2006, pp. 952-959.

[24] P. Thongtae and S, Srisuk, S. "An analysis of data mining applications in crime domain," in the International Conference on Computer and Information Technology Workshops, 2008, pp. 122-126.

[25] A. Kumar and A. Narain, "Designing of Controllers based on Artificial Neural Network for liquid Level System," in International Journal of Artificial Intelligence and Neural Network, Vol. 3, No. 3, 2013, pp. 1722.

[26] J. Caulkins, J. Cohen, W. Gorr and J. Wei, "Predicting Criminal Recidivism: A Comparison of Neural Network Models with Statistical Methods," in Journal of Criminal Justice, Vol. 24, No. 3, 1996, pp. 227240.

[27] F. Mingjian, Z. Guocheng, Z. Xuxu and Y. Zhongyi, "Study on Air Fine Particles Pollution Prediction of Main Traffic Route using Artificial Neural Network," in International Conference on Computer Distributed Control and Intelligent Environmental, 2011, pp. 1346-1349.

[28] X. Hong and R. Qunhua, "Flood Level Prediction on the Basic of the Artificial Neural Network," in

International Conference on Information Science and Engineering, 2009, pp. 4887-4890.

[29] Ahishakiye, Emmanuel & Taremwa, Danison & Opiyo, Elisha & Niyonzima, Ivan. (2017). Crime Prediction Using Decision Tree (J48) Classification Algorithm. International Journal of Computer and Information Technology (ISSN: 2279 – 0764).

[30] Olakorede, Mukaila. (2017). Principal Component Analysis of Crime Data in Gwagwalada Area Command, Abuja from 1995 – 2015. American Journal of Theoretical and Applied Statistics. 6. 38. 10.11648/j.ajtas.20170601.15.

Shubham Aggarwal

shubham.garg911@gmail.com +91- 99711 55172

Education

INTD. BTECH| GRADUATING BATCH OF 2021| COMPUTER SCIENCE ENGINEERING| JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA | CGPA 8.5 as per 7th semester.

12TH | CBSE - BATCH OF 2015 | MOTHER DIVINE PUBLIC SCHOOL, DELHI | 89.5%

10TH | CBSE - BATCH OF 2013| MOTHER DIVINE PUBLIC SCHOOL, DELHI|90%

Skills

- Operating Systems: Window.
- Programming Languages: C, C++, Python, HTML, CSS, Bootstrap, MySQL.
- Others: MS Office, Data Structure and Algorithm.
- Mathematics: Linear Algebra, Calculus, Basic Numerical Methods, Differential.
- Front End Website Development.
- Good Analytical, Managerial and Problem Solving Skills.

Minor Projects

MINOR 1| LIVE SCRIBBLING ECHO DIGI PEN | 07/2018 - 12/2018

- This project was to develop a Pen that will convert your content written by that pen into corresponding text file(.pdf).

MINOR 2| Restaurant Recommendation System and Site Selection | 01/2019 - 05/2019

- A Software designed to help customers select the best restaurant according to his preference and other parameters. It helps owners to get the potential customers according to the restaurant requirement. Also, locating a new best site for a person to install a new restaurant.

MAJOR 1&2| CRIME PREDICTION AND ANALYSIS FOR CHICAGO CITY | 07/2019 - PRESENT

- Crime prediction and analysis is not an easy approach for government officials. Several prediction systems have been developed in solving crimes and predicting the crime for a city. But, most of them lacks in output accuracy, reliable and quick.

Work Experience

SUMMER INTERNSHIP| AUTONOMOUS STARTUP ENTREPRENEUR | 06/2019 - 07/2019

- Worked there as a Web Developer Intern. Designed 2 Front End Webpage for 2 startups. Also, Help the Company in designing its own Webpage for an Event organized by ASE. Learned Back End development and various other Techniques and Styling used in front end Development.

Other Projects

- JIIT Hub Website - A place where fresher can enroll themselves and learn new things.
- Hangman Game made using Linked List and Dictionary using Hashing.
- Project on Automata Editor in Python.

Roles in College

- Student Volunteer: Microcontroller Based systems and Robotics Hub (uCR).
- OC Team: International Conference on Contemporary Computing (2018).
- OC Team: Cyber Srishti (Technical Fest of College) 2018
- Head of Management: Cyber Srishti (Technical Fest of College) 2019.

Volunteer Experience with NGOs

- Member of Rotaract Club of Outstanding Business Students (Rotary International) – **2018.**
- Secretary of Rotaract Club of Outstanding Business Students (Rotary International) – **2019.**
- Member of Sawariya Aahar Foundation - **2019 to present.**

Additional Details

- Paytm Mall Campus Ambassador 2018.
- Member of Google Developer Group, New Delhi.

Swaraj Saxena

Hno 3/38, Rail Vihar Colony Ph-2, Medical Road.
Gorakhpur, Uttar Pradesh, 273013
swarajsaxena10@gmail.com
+918574930145

QUALIFICATIONS

- **Jaypee Institute Of Information Technology** Integrated B.Tech-M.Tech in Computer Science with 6.3 CGPA till 5th semester 2016-2021
- **Little Flower School** 12th from ISC Board with 86.2% 2015
- **Little Flower School** 10th from ICSE Board with 89.6% 2013

TECHNICAL SKILLS

Core Java	★★★★☆
JavaScript	★★★☆☆
HTML	★★★☆☆
CSS	★★★☆☆
MySQL	★★★☆☆
LEX -YACC	★★★☆☆
Python	★★★★☆
C/C++	★★☆☆☆

PROJECTS

1) Cryptojacking : Attack vectors and obfuscation

- Port forwarding - Ngrok
- IDE - Brackets
- Browser - Chrome
- Custom Obfuscator - Python

2) Partial Digest Problem

- Python- Biopython

3) CarPooling System

- C++
- C

4) Parola Webpage(Frontend + Backend)

- HTML
- CSS
- PHP
- JavaScript

AWARDS AND ACCOMPLISHMENTS

- Football Team Captain
- 2nd in Manual Robot Competition
- School Sports Captain

Parinay Prateek

B.Tech Undergraduate

parinayprateek@gmail.com
+91-8130724229
Jaypee Institute of Information Technology,Sector-62, Noida, Noida, India

Result-oriented B.Tech Undergraduate who likes to take initiative and seek out new challenges. Have an inclination towards Cyber Security. Willing to adapt myself and possess eagerness towards learning thereby contributing towards the betterment of the organisation

EDUCATION

Marticulation
DAV Public School
2014 — *CGPA: 10*

Intermediate
S.N Sahay College
2016 — *Percentage: 73.6*

Bachelor of Technology in Computer Science and Engineering
Jaypee Institute Of Information Technology, Sector-62, Noida
2016 – Present — *CGPA: 6.9 (till 6th Sem)*

WORK EXPERIENCE

Summer Industrial Training
E & ICT Academy, IIT Kanpur
06/2019 – 07/2019
Electronics & ICT Academy (E & ICT Academy) at IIT Kanpur was established in 2016 in partnership with the Ministry of Electronics and Information Technology (MeitY), Government of India. It is mandated to provide industry focused and industry-driven hands-on courses in electronics & ICT. E & ICT Academy strives to narrow the gap between the academic approach to electronics and ICT domains as currently provided by the educational institutions and the practically oriented approach as demanded by the industry.

Tasks
- Penetration Testing on Web Servers

Global Internship
TakenMind
04/2019
TakenMind Global Internship Program is recognized under United Nations Sustainable Development and Growth (SDG) and is a highly recognized International Certification Program.

Tasks
- Data Analytics and Visualization

SKILLS

C programming | C++ programming | Python programming | Data Analytics | Cyber Security | Networking

PERSONAL PROJECTS

Penetration Testing on Web Servers
(06/2019 – 07/2019)
- This project aims to harden the security of company website and also secure employees from being social engineered. That requires a lot of footprinting and reconnaissance and hacking techniques. The main aim was to penetrate into the website. Various scans were carried out to find the website's vulnerability. The penetration testing was performed on vulnerable websites which are open for learning cyber security. Report was made on all the findings and security measures were proposed.

HealthCare Analysis On Dementia
(01/2019 – 05/2019)
- This project aims to provide a comprehensive study of comparing existing classification models and related tools when applied on a supervised dataset and to illustrate the potential of data analytics for different research areas in a similar fashion. This study focuses on the various attributes from the data that the authors collected, which help in identifying the possibility of a person suffering from dementia. The trend of each variable was generated with every other factor to help with the process of feature selection which in turn tracks the possible occurrence of dementia in a person. The authors trained the dataset using selected classifiers like Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, Gradient Boosting and AdaBoost. These classifiers were tested on our pre-processed dataset, and also on a self-created random dataset.

CERTIFICATES

Programming, Data Structures and Algorithms using Python
From NPTEL

Data Analytics Course
From Udemy

Microsoft Technology Associate
Networking Fundamentals

INTERESTS

Calligraphy | Playing Volleyball | Sketching | Cooking