**UNIVERSITY OF THE PHILIPPINES**

**Bachelor of Science in Computer Science**



Spatio-Temporal Prediction of Crime in Cebu City

**SP Adviser:**

**Prof. Demelo Lao**

**Department of Computer Science**

**University of the Philippines Cebu**

**Submitted by:**

**Jasper J. Bernales**

May 17, 2016

Contents

[Chapter 1 5](#_Toc452579283)

[INTRODUCTION 5](#_Toc452579284)

[1.1 BACKGROUND 5](#_Toc452579285)

[1.2 STATEMENT OF THE PROBLEM 7](#_Toc452579286)

[1.3 Objectives 8](#_Toc452579287)

[1.4 Significance of the Study 8](#_Toc452579288)

[Chapter 2 9](#_Toc452579289)

[Review of Related Literature 9](#_Toc452579290)

[Chapter 3 17](#_Toc452579291)

[Study Framework 17](#_Toc452579292)

[3.1 Theoretical Framework 17](#_Toc452579293)

[3.2 Operational Framework 18](#_Toc452579294)

[Chapter 4 19](#_Toc452579295)

[Methodology 19](#_Toc452579296)

[4.1 Creating Map Grid 20](#_Toc452579297)

[4.2 Data Collection and Preprocessing 20](#_Toc452579298)

[4.3 Modelling 22](#_Toc452579299)

[4.4 Statistical Analysis 31](#_Toc452579300)

# Chapter 1

## INTRODUCTION

### 1.1 BACKGROUND

Crime is one of the most horrible things that happen almost everyday especially in places with bigger population.

As of 2013, the United Nations Office on Drugs and Crime (UNODC) reported an increasing trend of crime (especially homicide, rape, theft, robbery and kidnapping) rate in the Philippines since 2009. This information is very alarming as it gives a bigger impact on victims and involves tremendous national social and economic costs.

Crime has been a major issue in the world today. The police officers who are the main people tasked to prevent and/or reduce crime sometimes feel like playing hide-and-seek - they suppress to decrease illegal activity in a high-crime area, then it’s already late to know that it happened somewhere else. They determine a crime hot spot, and then monitor that location, and they realize that it emerges in other places and then comes back the moment they leave.

To prevent crime, it is essential to recognize that a crime risk exists and corrective action is needed to eliminate or reduce the risk. However, police’s crime prevention is constrained by scarce resources (e.g. patrol units, finances and time). In order to supplement it, computer systems were used to identify and visualize areas with high crime or otherwise known as hotspots.

Several studies investigate on crime prediction through the use of previous crime data. Crime prediction gives an extension to the so-called hot spot methods and approaches of policing that recognize the criminality of certain places.

Since most research dwell on spatial analysis in modelling crime, this study will investigate whether it is possible to model crime through spatio-temporal data mining.

In the Philippines, crime is classified into two categories as defined by Philippine National Police for statistical analysis and for standardized definition of crime classification (Crimes Statistics at a Glance, 2013). These are index and non-index crimes. Index crimes are crimes against persons or property which includes murder, homicide, physical injury, rape, robbery, theft, carnapping/carjacking and cattle rustling. In contrast, non-index crimes are violation of special laws or local ordinances such as illegal logging.

In this study, we only focus in index crimes.

### 1.2 STATEMENT OF THE PROBLEM

Solving crimes has been the prerequisite of the criminal justice and law enforcement specialists. Computer systems and crime data analysis has been helpful in the last decades in solving and preventing crimes in several places of the world.

With the increasing population in Cebu City and development of more economic activity, its vulnerability to crime is also becoming more of a concern. The Police Regional Office of Central Visayas, which includes Cebu City, reported that there is an increase in crime volume in the region as of August, 2015.

On the other hand, Senior Supt. Rey Lyndon Lawas, the PRO 7 directorial staff chief, said that the intensified patrols and security operations which were conducted by the police are very useful in deterring crimes (Silva, 2015). With this, having an idea of when and where a crime would likely to happen will be more of a use in deploying police resources especially in a large city with very big population.

Although there is already a study regarding the prediction of crime in Cebu City, further research is still needed to supplement.

To enhance the previous researches in the prediction of crime in Cebu City, the current research would like to investigate on machine learning techniques that could predict crime in an area in a specific span of time. These predictions could be achieved using data on historical crime focusing more on a time-series data.

In order to answer the main problem, the study should answer the following question:

Is it possible to model a spatio-temporal prediction of crime in Cebu Citiy using data on previous crime as a time-series data?

### 1.3 Objectives

The following will be the objectives of this study:

Determine if it is possible to predict occurrence of crime in Cebu City using spatio-temporal forecasting in a grid thematic map.

If so, find out the smallest area of grid in the thematic map and smallest day interval possible to predict crime.

### 1.4 Significance of the Study

This study aims to investigate if it is possible to develop a crime prediction model through the use of previous crime data as a time series data.

The results of this study will be helpful to the police in their plans and strategies to combat crime. Smart, effective, and proactive policing is clearly preferable to simply reacting to criminal acts. Police deployment will be more efficient when there is information about the risk of an area to be a spot for crime occurrences. Consequently, crime rates can be lowered if not avoided.

Moreover, the framework of this study can also be used for future studies on crime prediction and time series prediction.

# Chapter 2

## Review of Related Literature

#### Techniques in Predicting Crime

McClendon et. al (2015) conducted a study entitled “Using Machine Learning Algorithms to Analyze Crime Data” to prove how effective and accurate the machine learning algorithms used in data mining analysis can be at predicting violent crime patterns.  The dataset consisted of socio-economic data from the 1990 Census, law enforcement data, and crime data. The dataset also consisted 2215 instances or crimes that had been reported from across the country and 147 total attributes, commonly referred to as features. The features analyzed were the murders, murder per population, rapes, rapes per population, robberies, robberies per population, assaults, assault per population, and violent crimes per population where per population equals to 100,000. The proponents used an open source data mining software called WEKA to implement and conduct analysis on the dataset. The algorithms used were Linear Regression, Additive Regression, and Decision Stump which were all provided by WEKA. After implementation of the algorithms, WEKA outputs five metrics that evaluate the effectiveness and efficiency of the algorithms: Correlation coefficient, Mean absolute error, Root mean squared error, Relative absolute error, and the Root relative squared error. The results for these five metrics were used in the comparative evaluation of the crime statistics.  The results showed linear regression to be very effective and accurate in predicting crime data based on the training set input of three algorithms. The decision stump algorithm achieved a relatively poor performance because of a certain factor of randomness in the various crimes and the associated features. The additive regression model seemed to be most prominent for all of the crimes per 100,000 population features except for the RobberiesPerPop, AssaultsPerPop, and ViolentCrimesPerPop features.

Adams (2001) conducted an exploratory spatial analysis of historical homicide hot spots in three cities of Columbia. The study also took into account the possible underlying factors associated with the emergence and persistence of homicide hotspots. Logistic regression was performed to assess how well some of the social, economic and built environment factors were able to predict the presence or absence of the historical homicide hot spots. It was found out that the social and economic factors appear to have a significant impact on the emergence of homicide hot spots.

Moreover, economic indicators that predicts property and violent crime rates were also studied by Alwee et. al(2013). A combination of grey relational analysis and support vector regression were used to establish a model that can rank and select the significant economic indicators for crime rates. Support vector regression is a nonlinear model to solve regression problems. On the other hand, grey relational analysis uses data series to obtain grey relational order to describe relationship between related series. Results of the study showed that these methods produced smaller errors compared to multiple linear regression when it comes to forecasting property and violent crime rates.

A study called Olligschlaeger (2015) aimed to predict the emergence or "flare ups" of drug hot-spot areas in the city of Pittsburgh. The dataset used were drug related 911 calls recorded by Pittsburg (PA) Bureau of Police. The proportion of residential to commercial property was also included as an indicator variable.  A map of the city of Pittsburgh with all the call data was then divided into a grid with a size of 2,150 square feet for each cell. In each cell, the number of calls per month was obtained. Three methods were used to estimate the "one-step-ahead" forecasting model. Namely, ordinary least squares regression with six independent variables for each cell in the neighborhood, neural network similar to the "Game of Life" neural network with constant weights, neural network with spatially varying input-to-hidden unit weights. Data from 1990 to 1991 was used to estimate regression parameters as the training set. The 1992 data were used as the test set. Learning rate was set to 0.0001 with maximum of 15000 iterations. The results showed that the regression model and constant weight network performed similarly. The varying weight model did significantly better at modelling the hotspot areas compared to real actual hotspot areas. The proponent suggested to modify feedforward and backpropagation neural network to converge quickly to a solution.

Meanwhile, a similar machine learning technique, Artificial Neural Network, was conducted by Corcoran, J. et. al(2003) in a study called Predicting the Geo-temporal Variations of Crime and Disorder to test an algorithm in predicting crime. The dataset, which spanned for 1 year, consisted 18, 498 violent incidents which includes: violence against the person, criminal damage, and disorder. There were 3 main stages in the technique used in this study. The first (spatial analysis) identified geographical clusters with point density analysis; the second (cluster modelling) determines the data quality of each cluster with Gamma Test; the third (prediction) develops a corresponding artificial neural network (ANN) model based on an autoregressive predictive specification. The proponents compared the performance of their technique with linear regression and modified random walk. The output of the Gamma Test was also used as an input for the linear regression which was the same with the artificial neural network. The results showed that ANN generally models the trends in each cluster the best among the three algorithms used. For future improvements, the proponents suggest to evaluate other variables that could affect crime trends such as forthcoming public holiday and predicted weather.

A study titled “Short-term Forecasting of Crime” (Olligschlaeger A. et al, 2003) was conducted to investigate whether it is possible to accurately forecast selected crimes 1 month ahead in small areas, such as police precincts in Pittsburg, PA. The primary data consist of 8 years (approximately 1 million records) of 911 CAD (Computer Aided cincts 2 and 5, both containing large run-down Dispatch) and offense report data for all individual events for 1991–1998 obtained from the Pittsburgh poor neighborhoods. The types of crimes selected to study were robbery, burglary, simple assault, aggravated assault, and drug activity.  All records were mapped by address using GIS, yielding points on a street map. Point data were spatially aggregated into monthly time series of crime counts by precincts, producing 30 univariate time series, one for each seven precincts and for five crime types. Classical decomposition was used to calculate monthly seasonal indices at both the city and precinct level with multiplicative ratio-to-moving averages so that the dataset would be are dimensionless, allowing comparisons across geographic areas of differing crime scales. For each 30 time series, two deseasonalized series were generated, one series calculated using the individual precinct’s seasonal indices, and the other series using the pooled city seasonal indices, calculated using the relevant data sample for all of Pittsburgh. The proponents forecast a total of 100 series (30 unadjusted series, 30 precincts series deseasonalized using precinct-level seasonal factors, 30 precinct series deseasonalized using pooled city-wide seasonal factors, five unadjusted city-wide series, and five city series deseasonalized with city-wide seasonal factors).  Only the data before the forecast origin, in the preceding year, was used in decomposition. The seasonal indices were recalculated annually for each series (three times, once for each year), using a rolling-window of 5-years periods (1991-1995, 1992-1996, 1993-1997). Each of the 100 series was applied with 3 forecasting methods: Random Walk, Brown’s Simple Exponential Smoothing, and Holt’s Two-parameter Linear Exponential Smoothing. An optimization procedure was used to estimate the smoothing parameter. After the forecasts were made, the forecasts were reseasonalized to allow comparison with actual values in computing forecast errors. For each monthly forecast, the immediately preceding 60-month period was used to estimate the forecast model, and the forecast was then compared to the actual holdout month to calculate the resultant error. Thus, February 1996 was based from the data from the period February 1, 1991 through January 31, 1996. Rolling the window 1 month forward, the forecast for March 1996 was based on March 1, 1991 to Febuary 29, 1996, and so on. The overall results showed that random walk performs poorly and it is better to use city-wide seasonal indices with either Holt or simple exponential smoothing.

Another approach in modeling crime as suggested by Heim(2014) is through analysis over street segments. Street segments were used as the basic study unit in developing crime hotspot analysis. Factors were social disorder, age, elevation, count of males and housing density. Through zero-inflated Poisson, Heim(2014) crime data that have an excessive number of zero counts were modeled separately. Crime counts were then smoothened over the road segments in order to better visualize the occurrence of crime hotspots. The resulting principal components analysis of road segment shows that different types of variables can describe crime in different locations.

On the other hand, grid-thematic mapping and artificial neural network approach was used in modeling and forecasting crime in Cebu City(“Crime Modeling and Prediction Using Neural Network”, 2014).

Previous works on crime prediction focused heavily on factors affecting crime rates. Although some studies also touched on predicting crime using machine learning techniques, most of their datasets were only treated as an input-output data. It was suggested to treat the crime data as a time-series data that would predict future crime counts for each cell in the grid in order to come up with a better model.

Thus, this study will focus on time-series prediction of crime. With the limited data available, basic crime data including the classification of data, date of occurrence and location will be the dataset to be used in this study.

#### Algorithms for Time-Series Prediction

Recently, Deep Nueral Networks have been successful in spatio-temporal prediction (Shi et.al., 2015), recognition ([Donahue et.al., 2014](file:///C:\Users\maebernales\Documents\thesis\references\long-term%20recurrent%20convolutional%20networks%20for%20visual%20recognition%20and%20description.pdf)) and classification (Krizhevsky et. al., 2012).

According to the book entitled “Deep Learning for Java”, deep-learning networks are distinguished from the more common single-hidden-layer neural networks by their depth; that is, the number of node layers through which data passes in a multistep process of pattern recognition.

In addition, it explained that traditional machine learning relies on shallow nets, composed of one input and one output layer, and at most one hidden layer in between. More than three layers (including input and output) qualifies as “deep” learning. So deep is a strictly defined, technical term that means more than one hidden layer.

In deep-learning networks, each layer of nodes trains on a distinct set of features based on the previous layer’s output. The further you advance into the neural net, the more complex the features your nodes can recognize, since they aggregate and recombine features from the previous layer.

David et. al.(2014) experimented on the use of Spiking Neural Networks in financial time-series prediction. The said algorithm was able to predict the future price of crude oil with historical data on IBM stocks, US/Euro exchange rate, and price of crude oil.  Results revealed that Spiking neural network is effective for non-stationary data which means that the statistical properties, such as the mean and variance, of the data changes over time.

Similarly, Jha & Sinha(2012) compared time-delay neural networks(TDNN) and autoregressive integrated moving average (ARIMA)  in predicting wholesale price of oilseeds in India. The results suggested that the TDNN model in general provides better forecast accuracy in terms of conventional root mean square error values as compared to ARIMA model for nonlinear patterns. The study also reveals that the neural network models have clear advantage over linear models for predicting the direction of monthly price change for different series. Moreover, the study suggests that before adopting any nonlinear model, one needs to check whether the series is indeed nonlinear.

On the other hand, Wang and Cheng (2008) used Integrated Spatio-Temporal data mining in predicting forest fire in Canada. They study two provinces and conducted first a time-series analysis using ARIMA (autoregressive integrated moving average) model. The results of the temporal forecasting were used in spatial forecasting through Elman recurrent neural network. Spatial and temporal forecast were combined through linear regression to come up with an overall forecast of forest fire.

Recently, the state-of-the-art deep neural network for spatio-temporal prediction is the Convolutional LSTM(Long Short-Term Memory) Network. In a study conducted by Shi et. al.(2015), it showed that Convolutional LSTM network “captures spatiotemporal correlations better and consistently outperforms FC(fully-connected)-LSTM”. Further, results revealed that using Convolutional LSTM network, future rainfall intensity in a local region was predicted over a relatively short period of time.

For this study, Convolutional LSTM network will be used for the spatio-temporal prediction of crime in Cebu City.

# Chapter 3

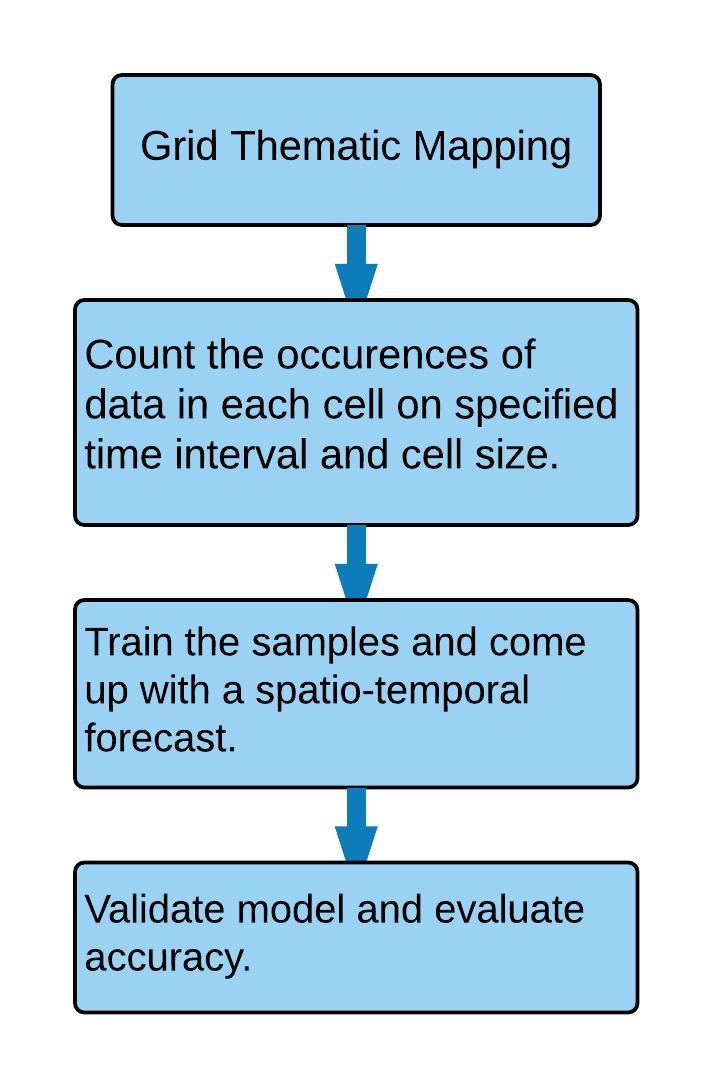
## Study Framework

This section discusses the different frameworks that will be used in predicting crime areas. These will give an overview on how the study will be conducted and which theories they are anchored to. It will be divided into two parts: the theoretical framework and operational framework. The theoretical framework presents the theories that support the fundamental concept of the study while the operational framework gives an overview of the specific processes to be conducted in the study based on the theories presented in the theoretical framework.

### 3.1 Theoretical Framework

The theoretical framework of this study shows the general idea where the study will be based.

In the work entitled “Crime Modelling and Prediction Using Neural Networks”(2015), grid thematic mapping was used to avoid the problem of varied shapes and size to be able to have a comparable and consistent comparisons with each areas of interest. Moreover, related studies on spatio-temporal prediction and mapping (Shi et.al., 2015, David et. al., 2014, Jha & Sinha, 2012, Wang & Cheng, 2008) used datasets to apply machine learning approaches in coming up with a spatio-temporal model using a time-series data. Lastly, model validation and accuracy evaluation were also conducted for every machine learning algorithm being suggested.

­­

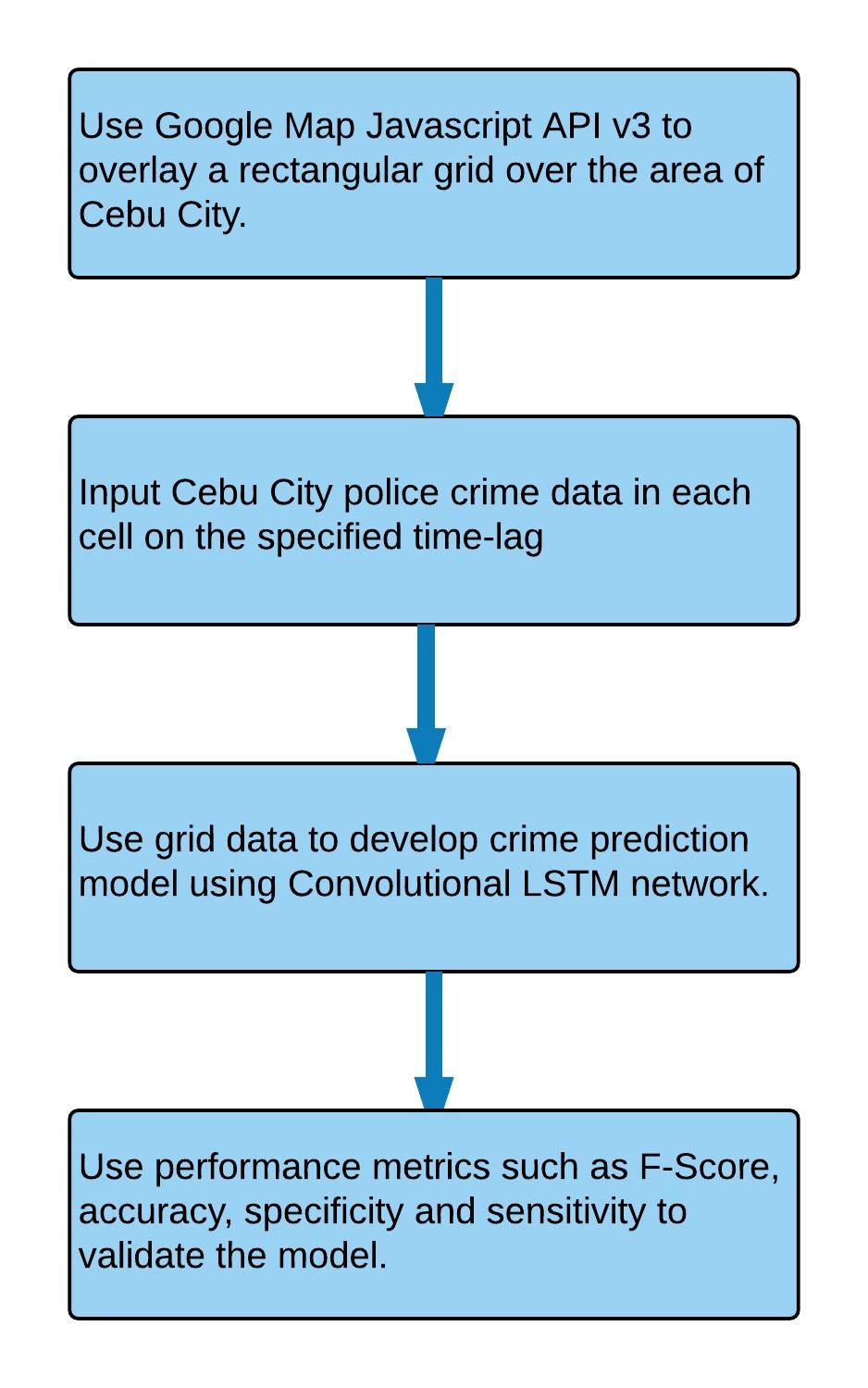
**Fig 1. Theoretical Framework**

The figure above shows the theoretical framework of this study.

### 3.2 Operational Framework

Google Maps provides developer features useful in overlaying square grids and crime data into a map. Number of crime incidents in each cell can automatically be generated over each day or week using the created grid. Moreover, data can be exported into a text file to be used as inputs on convolutional LSTM network for spatio-temporal forecasting(Shi et. al., 2015). According to Shi et. al.(2015), the convolutional LSTM determines the future state of a certain cell in the grid by the inputs and past states of its local neighbors. This can easily be achieved by using a convolution operator in the state-to-state and input-to-state transitions.

Lastly, the model developed based upon training data will be evaluated with the testing data. Performance metrics will be used to validate the model.



**Fig 2. Operational Framework**

The figure above shows the operational framework of the study.

# Chapter 4

## Methodology

This section will present the methods that were used in this study.

### 4.1 Creating Map Grid

Replicating the strategy used in creating a map grid by previous study (Crime Modelling and Prediction Using Neural Networks, 2015), Google Maps JavaScript API v5. The area of interest (i.e. Cebu City) was defined by creating a polygon of its bounds. However, cell merging was not implemented because the proponent focused on the urban area only. The researcher experimented on different grid sizes to come up with the smallest possible grid that still has a high accuracy and F-Score. The figure below shows the map grid for Cebu City which is the area of interest in this study:

### 4.2 Data Collection and Preprocessing

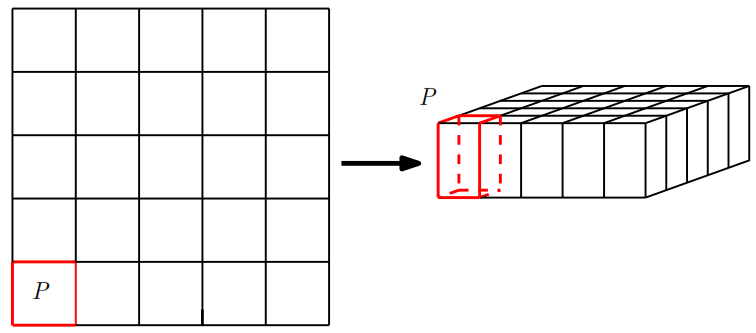
Crime report data will be collected from the Cebu City Police office. The dataset will be from the Crime Reporting System Form 1 (NCRS Form 1) which contains various information of crime incidents. The dataset spans for 3 years from the months of January 2014 to December 2015. The fields that are useful in this study are the date the crime was committed, type of crime, address of the committed crime including the latitude and longitude. Using the created grids, number of crime incidents in each cell can automatically be generated over each day or week through the Google Maps JavaScript API.

Before using the dataset for modelling, it was first normalized using standard score(Z-Score) in order to lessen training duration.

Since this is a spatio-temporal problem, the weekly data was transformed into a rolling window with stride 1 and length n. This means that it would be a time-series of 2-dimensional data. The data is described as follows:

Suppose we have *X* as a single time-series example and *n* as the number of observed values, the input for the model consists of *X1, X2, . , . ,Xn-1* and outputs *X2, X3, . , . ,Xn.* In this case, we are only interested on the last observation of the model’s output, which is *Xn*. Thus, the dimension of the input and output is *n* x M x N. In the figure below, P represents the observation for a single grid over time.

Number of Time-series examples = ((All weekly examples-n)/Stride)+1



The dataset was divided into 60:20:20 ratio. This means that 60% of the data would be for training, 20% for validation and the remaining 20% would be for testing. The validation dataset would be used to check if the model has over fitted or not. On the other hand, the testing dataset would be used to assess the result of each input.(Ng, 2016)

### 4.3 Modelling

This section explains the implementation of recurrent neural network in crime occurrence prediction based from Graves(2014) methodology for hand writing recognition.

#### RNN and LSTM

Recently, recurrent neural networks are rich class of dynamic models that have been very useful to produce sequences in a wide scope of domains such as music, text and motion capture data. RNNs can be trained for sequence generation by processing real data sequences one step at a time and forecasting the future sequence.

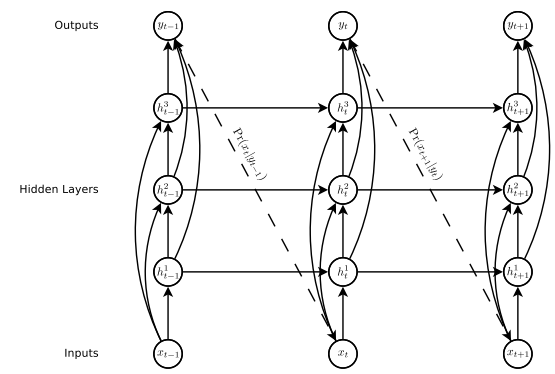
RNNs do not use exact templates from the training data to make forecast, instead, they use their internal representation to perform a high-dimensional interpolation between training examples. However, the disadvantage of standard RNNs is that they are unable to store information about past inputs for very long which makes them prone to instability when producing sequences. The network’s predictions are only based on the last few inputs and it has a little opportunity to recover from past mistakes. Thus, a longer memory will compensate for this, because even if the network cannot make sense of its recent history, it can go back further in the past to formulate its predictions.

The problem of instability is only acute with real-valued data, where it is easy for the predictions to stray from the manifold on which the training data lies. One solution that has been suggested is to inject noise into the predictions before feeding them back into the model, thus increasing the model’s robustness to surprising inputs.

According to Graves(2014), Long Short-term Memory (LSTM) is an RNN architecture aimed to be better at storing and accessing information than standard RNNs. It has recently produced “state-of-the-art results” in variety of sequence processing tasks such as speech, handwriting recognition and precipitation nowcasting (Shi et. al., 2015; Graves et. al., 2013;Graves & Schmidhuber, 2008). Results from previous studies revealed that LSTM can use its memory to generate complex, realistic sequences containing long-range structure.

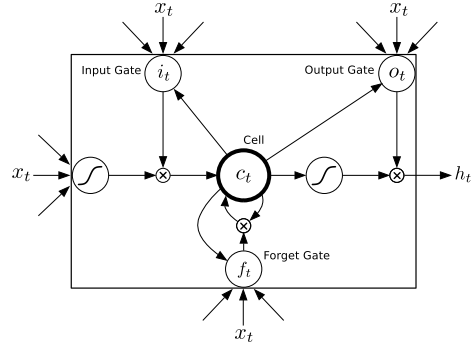
The key equations for LSTM are shown below, where ‘°’ signifies the Hadamard product:

The figure below illustrates the deep recurrent neural network architecture. The circles represent the network layers(LSTM cell), the solid lines represent weighted connections and dashed lines represent predictions.



Deep recurrent neural network architecture.

On the other hand, the image below shows the structure of an LSTM cell.

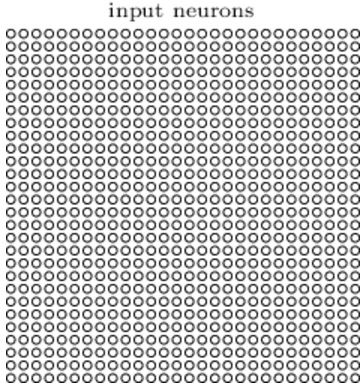


LSTM cell

#### Convolutional Operation

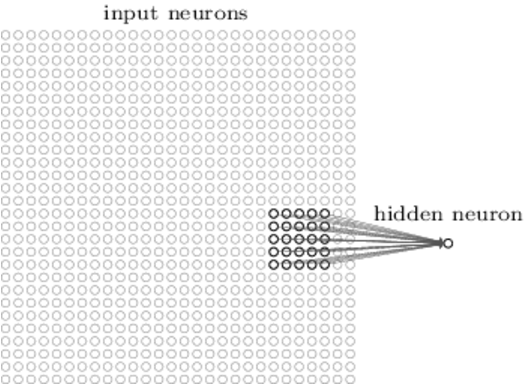
Local Connectivity

For this study, the values of each neuron correspond to the grids in the map that we’re using as inputs. Thus, a 28x28 square of neurons correspond to the 28x28 grids in the map.



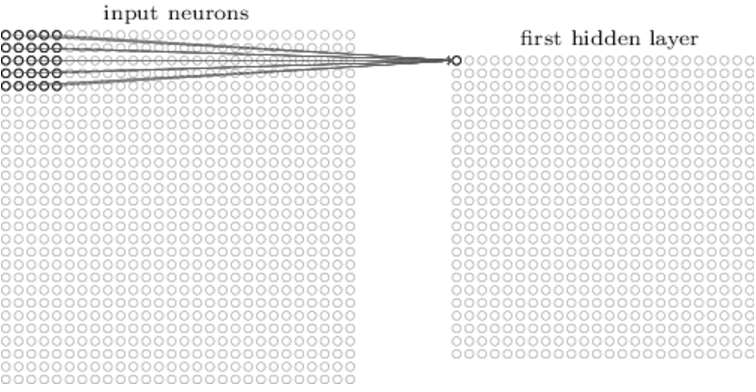
The input grids were connected to a layer of hidden neurons but not all grids were connected to every hidden neuron. Instead, the connections were only in the small, localized regions of the input grid.

This means that each neuron in the hidden layer will be connected to a small region of the input neurons. For example, a 5x5 region, corresponding to 25 input grids. So, for a particular hidden neuron, we might have connections that look like this:

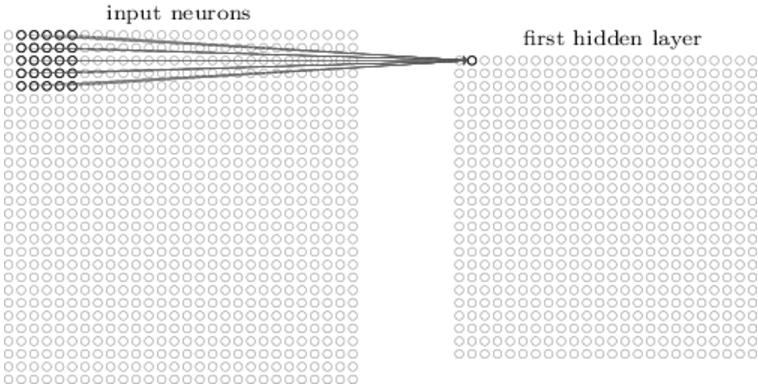


The region in the input image is called the *local receptive field* for the hidden neuron. It’s a little window on the input grids. Each connection learns a weight and the hidden neuron learns an overall bias as well. That particular hidden neuron would be learning to analyze its particular local receptive field.

The local receptive field were then slide across the entire input grid. There’s also a different hidden neuron in the first hidden layer for each local receptive field. The figure below shows how local connectivity is implemented starting with the first 5x5 grid from the top left corner.



Then the local receptive field were slide over by one grid or neuron (as shown in the image below), to connect to a second hidden neuron.



The process goes on building up the first hidden layer. Note that if we have a 28x28 input image, and 5x5 local receptive fields, then there will be 24x24 neurons in the hidden layer since the local receptive field can only be moved 23 neurons across, before colliding with the right-hand side (or bottom) of the input image.

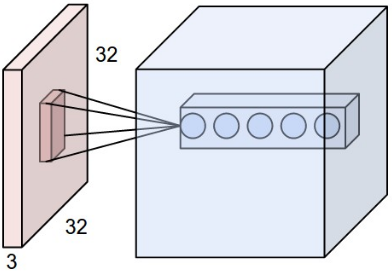
Convolutional Layer Spatial Arrangement

In this section, the arrangement of neurons in the output volume will be explained. There are 3 hyper-parameters that control the size of the output volume. These are depth, stride and zero-padding.

1. Depth- this corresponds to the number of filters that is used, each learning to look for something different in the input.
2. Stride – refers to the number of slide of the filter. When the stride is 1, then the filters are moved one grid at a time.
3. Zero-padding – the size of the padding of the input volume with zeros around the border. The advantage of this is that it will allow the control of spatial size of the output volumes.

Parameter Sharing

The filter(kernel) that was used for a single depth uses the same weights and bias. However, the filter for each is not the same. Therefore, there is only one filter used for each depth.

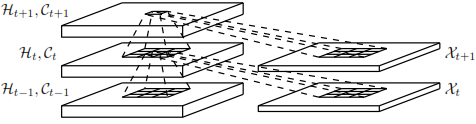


The image above shows an example of 32x32x3 input volume(left) with each neuron in the convolutional layer(right) connected spatially only to its a local region, but to all its color channels. Please note that there are multiple neurons along the depth, all looking at the same region in the input.

Convolutional LSTM

A major disadvantage of a standard LSTM in dealing with spatiotemporal data is the usage of full connections in input-to state transitions in which no spatial information is encoded. To solve this problem, Convolutional LSTM(ConvLSTM) has a design in which all inputs *X1,…Xt* cell outputs, *C1,...Ct,* hidden states *H1,...Ht* and gates *i1,ft,ot* of the ConvLSTM are 3D tensors whose last two dimensions are spatial dimensions. The input and states may be imagined as vectors standing on a spatial grid. The ConvLSTM decides the future state of a certain cell in the grid through the inputs and past states of its local neighbors using a convolution operator in the state-to-state and input-to-state transitions. The key equations of ConvLSTM is shown in Equation 2 where ‘\*’ denotes the convolution operator and ‘’ denotes Hadamard product:

The image below shows the inner structure of ConvLSTM.



Implementation

The generated file will be used for the training of ConvLSTM neural network model. It was implemented using Python and a library called “Keras”*.*

The results of the Convolutional LSTM neural network model were inputted back to the JavaScript map application in order to visualize the results. The application will show the likelihood of crime incidence for each cell. Darker colored cell would mean higher probability of crime incidence in that area while lighter colored cell would mean otherwise.

### 4.4 Statistical Analysis

Accuracy assessment of the model was determined using Binary Cross Entropy. Accuracy, sensitivity, specificity and F1 Score were used to check if the model accurately predicted a similar pattern to the actual data.

**Results**

**REFERENCES**

Adams, T. M. (2001). Historical Homicide Hot Spots: The Case of

Three Cities. Ann Arbor, MI:UMI.

Alwee, R., Shamsuddin, S.M.H., Sallehuddin, R.S(2013). Economic indicators selection for crime rates forecasting using cooperative feature selection. In *Proceeding of the 20th National Symposium on Mathematical Sciences Research in Mathematical Sciences: A Catalyst for Creativity and Innovation*, 1522, 1221–1231. DOI: 10.1007/978-3-319-16211-9\_11

Cheng, T., Wang, J. (2008). Integrated Spatio-temporal Data Mining for Forest Fire Prediction. Transactions in GIS, 12(5), 591-611. Retrieved from http://www.isprs.org/proceedings/XXXVI/part7/PDF/148.pdf.

Corcoran, J. J., Wilson, I. D., Ware, D. (2013). Predicting the geo-temporal variations of crime and disorder. International Journal of Forecasting, 19, 623-634. Retrieved from <http://principlesofforecasting.com/files/pdf/Corcoran_Wilson_and_Ware,_Predicting.pdf>.

Deeplearning4j Development Team. Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0. [http://deeplearning4j.org](http://deeplearning4j.org/)

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2625-2634).

Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Graves, A., Mohamed, A. & Hinton, G.(2013). Speech recognition with deep recurrent neural networks. In Proc. *ICASSP*.

Graves, A. & Schmidhuber, J.(2008). Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 21.

Heim, K.(2014, August). Visualization and Modeling for Crime Data Indexed by Road Segments (Doctoral Dissertation). Retrieved from http://digilib.gmu.edu/xmlui/bitstream/handle/1920/8991/Heim\_gmu\_0883E\_10696.pdf?sequence=1&isAllowed=y.

Jha, G., & Sinha, K. (2014). Time-delay neural networks for time series prediction: an application to the monthly wholesale price of oilseeds in India. Neural Computing & Applications, 24(3/4), 563-571. doi:10.1007/s00521-012-1264-z

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

McLendon, L., Meghanathan, N. (2015, March). Using machine learning algorithms to analyze crime data. Machine Learning and Applications: An International Journal, 44. Retrieved from <http://airccse.org/journal/mlaij/papers/2115mlaij01.pdf>.

Ng, A. (n.d.). Model Selection and Train/Validation/Test Sets[Lecture Notes]. Retrieved May 31, 2016, from https://class.coursera.org/ml-005/lecture

Olliigchlaeger, A. M.(1997). Artificial Neural Networks and Crime Mapping. Crime Mapping and Crime Prevention, 8, 313-347. Retrieved from <http://www.popcenter.org/library/crimeprevention/volume_08/11-olligschlaeger.pdf>.

Olligschlaeger A., Gorr, W., Thompson, Y. (2003). Short-term forecasting of crime. International Journal of Forecasting, 19, 579-594. Retrieved from <http://forecastingprinciples.com/files/pdf/Gorr_Olligschalger_and_Thompson,_Short-term.pdf>.

Reid, D., Hussain, A. J., & Tawfik, H. (2014). Financial Time Series Prediction Using Spiking Neural Networks. Plos ONE, 9(8), 1-13. doi:10.1371/journal.pone.0103656

Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *arXiv preprint arXiv:1506.04214*.

Silva, V. V. (2015, September 24). Crime volume increases in Region 7 as of Aug. *Cebu Daily News.* Retrieved from <http://cebudailynews.inquirer.net/>

Tumulak, J.(2015, May). Crime Modeling and Prediction Using Neural Networks.

UNODC Statistics. (2015). Retrieved from <https://www.unodc.org/unodc/en/data-and-analysis/statistics/crime.html>