Department Computer Science and Engineering.
Yuan Ze University, Taiwan.

Stock Market Prediction Using Deep Learning Neural Networks and Candlestick Chart

Master Thesis Defense

Advisor : Yu-Yen Ou

Student : Rosdyana Mangir Irawan Kusuma

Tuesday, July, 19th, 2018

OUTLINE

1

INTRODUCTION

Introduction

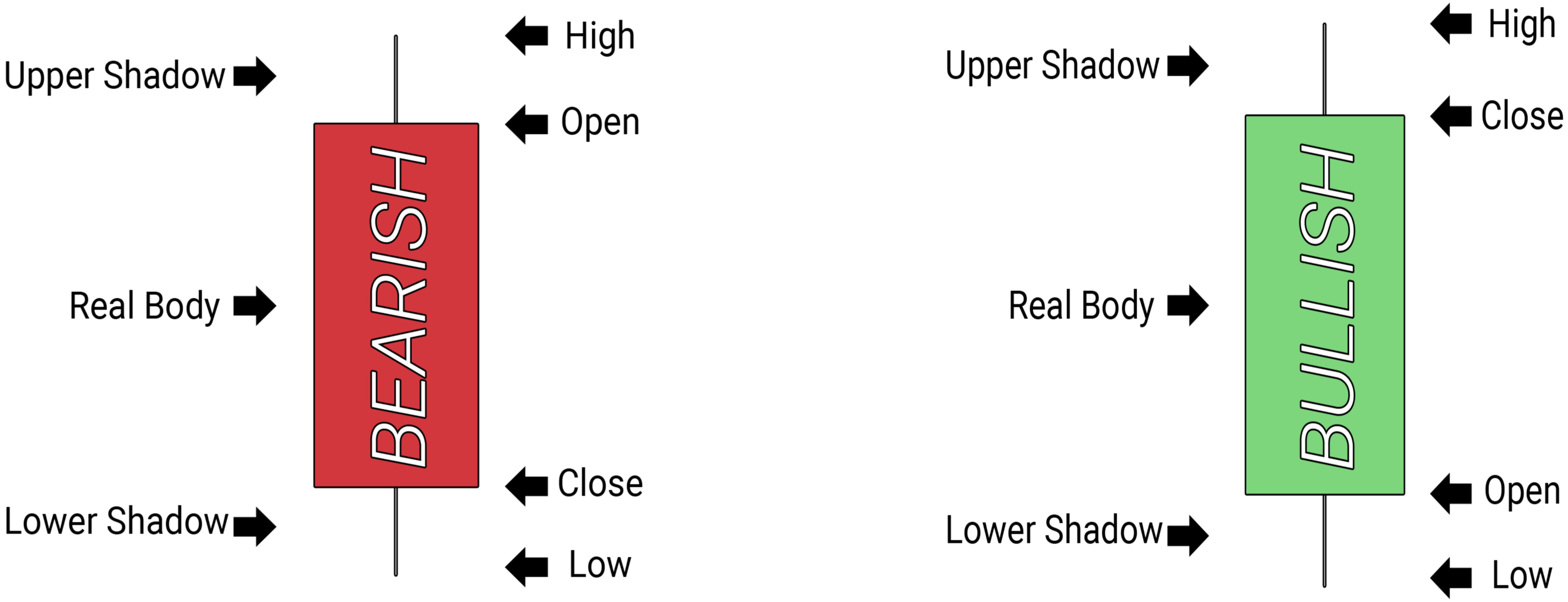
- A **stock market**, **equity market** or **share market** is the aggregation of buyers and sellers of stocks, which represent ownership claims on businesses.
- **Stock market crash** : Panic of 1907, Wall Street Crash of 1929, Black Monday 1987, Crash of 2008-2009

Motivations

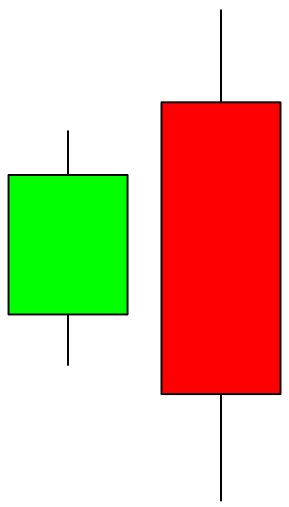
- Stock market prediction is a **challenging problem**:
  - Company news and performance
  - Industry performance
  - Economic factor
  - Social media sentiment
  - Investor sentiment
- Helps trader to **enhance** their information about **stock market movements**.
- Perform good prediction to get **more benefit** in stock market **trading**.

Candlestick Chart

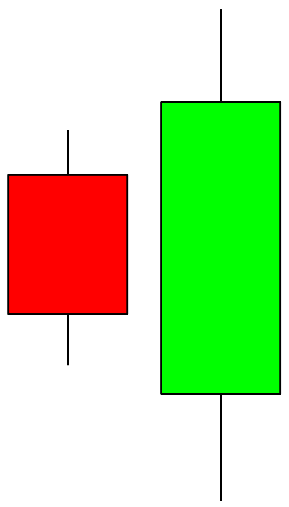
- **A style** of financial chart used to **describe price movements** of stock market.
- Contains **open, high, low** and **close price** value.
- **Candlestick pattern** is a **movement in prices** shown graphically on a candlestick chart that some believe can **predict a particular market movement**.
- Currently there are **41 recognised patterns**.



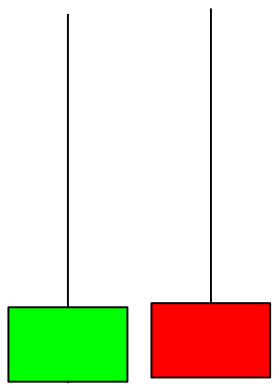
Candlestick Patterns



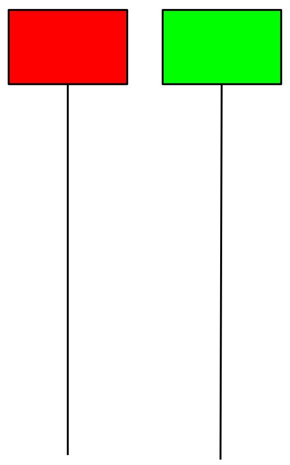
Bullish Engulfing



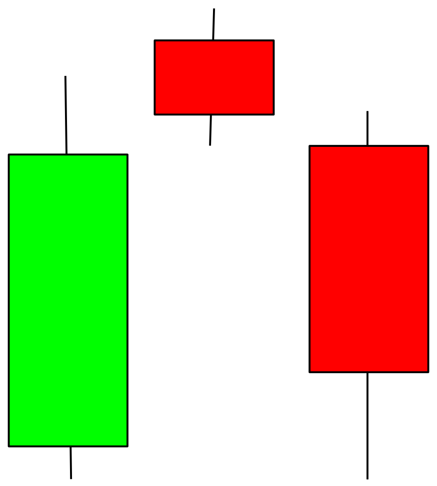
Bearish Engulfing



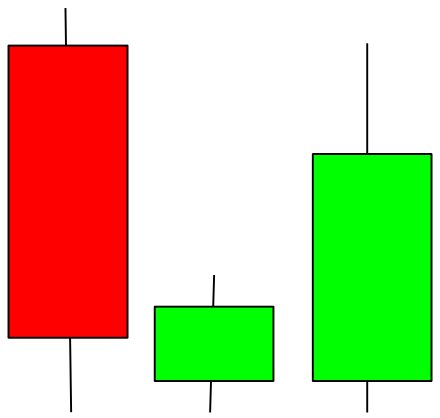
Shooting Star



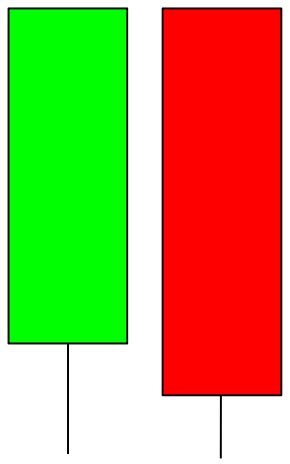
Hammer



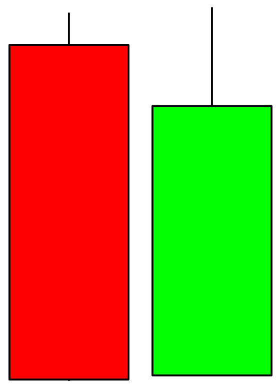
Evening Star



Morning Star



Bearish tweezer



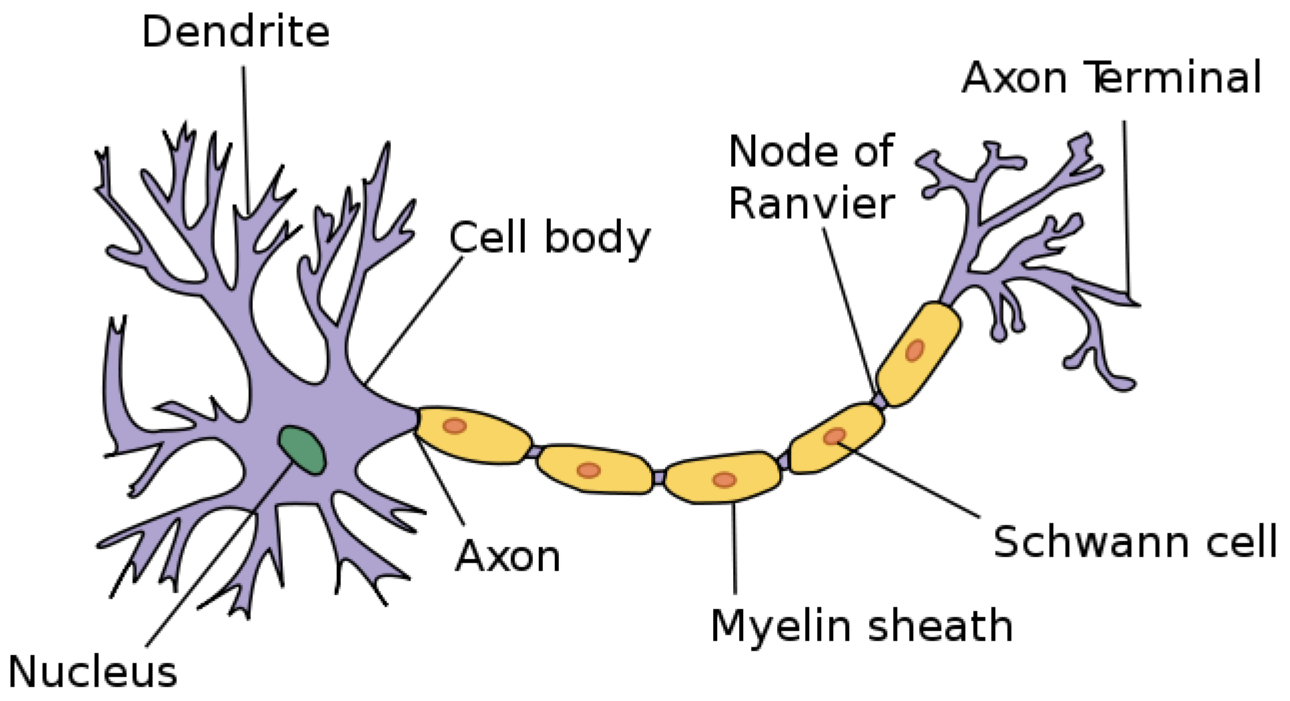
Bullish tweezer

Historical Stock Market Data

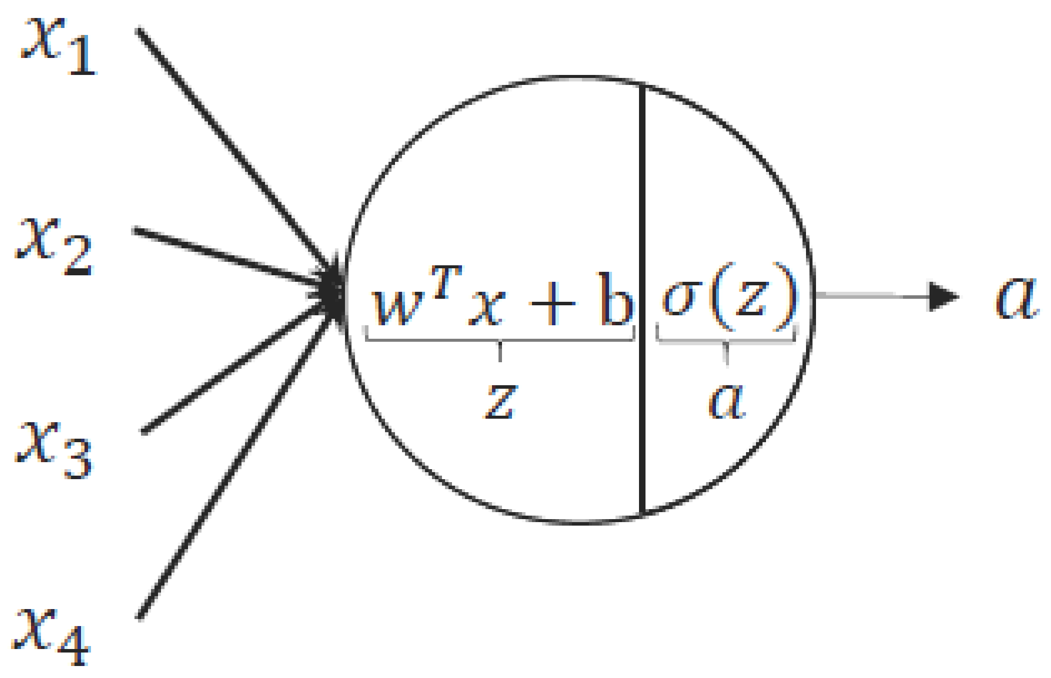
Name	Description
Open Price	The first price in daily activity of the stock market that is noted when the stock market opens in the specified period.
Close Price	The final price at which a security is traded on a given trading day.
High Price	Highest price at which a stock traded during the course of the day.
Low Price	Lowest price at which a stock trades over the course of a trading day.
Volume	the number of shares or contracts traded in a security or an entire market during a given period.

Deep Learning

❖ Using brain-inspired mechanics to achieve brain-like function



A Neuron in our brain



A Neuron in our neuron network

Traditional Learning Approach vs Deep Learning Approach

❖ ***Traditional Learning architecture***:

Input (Image, Text,..) → Hand-design feature extraction → Trainable classifier → Object class

❖ ***Deep Learning architecture:***

Input (Image, Text,..) → Layer 1 → … Layer N → Simple classifier → Object class

RELATED WORK

Related Work

Author	Idea and Method	Result
Patel, Shah et al. 2015	Added 10 technical parameter in feature set to perform stock market prediction using 4 algorithm (ANN, SVM, random forest, naive-bayes).	Highest result by naive-bayes with average 90 % accuracy.
Khaidem, Saha et al. 2016	Using random forest with adding RSI in 3 different trading data(APPL, GE, Samsung).	Average result 89 % accuracy
(Zhang, Zhang et al. 2018	Combine trading data with sentiment from social media and financial news in Hong Kong stock market.	Highest result 61 % accuracy.

METHODOLOGY

Methodology Design

Data Collection

Training : 2000-01-01 - 2016-12-31

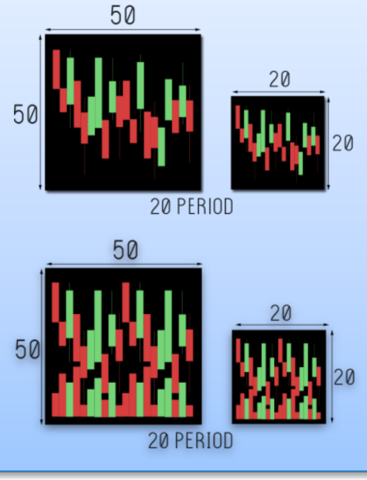
Testing : 2000-01-01 - 2016-12-31

Independent : 2000-01-01 - 2016-12-31

Name	Training	Testing	Independent
TW50	198866	14756	348
ID10	34458	3692	348

Data Preprocessing

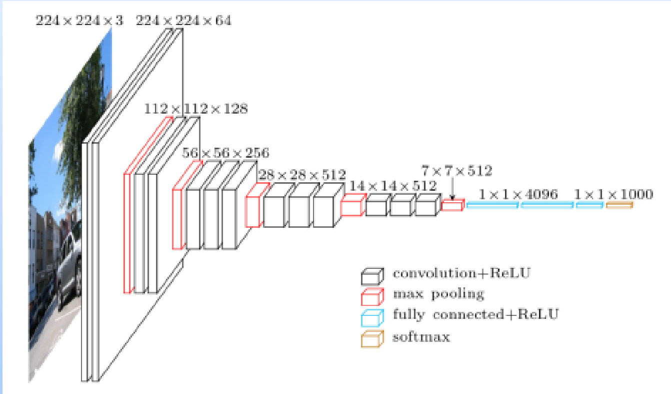
Convert to Candlestick chart



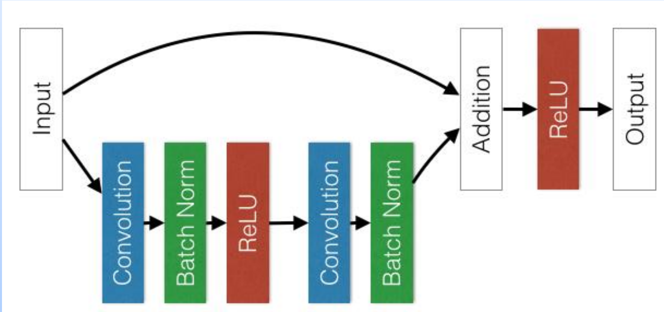
Sliding Window

	5 Periods			10 Periods			20 Periods		
Name	Training	Testing	Independent	Training	Testing	Independent	Training	Testing	Independent
TW50	198569	17164	342	198151	16950	337	197819	16414	327
ID10	34350	3611	342	34323	3582	337	34233	3482	327

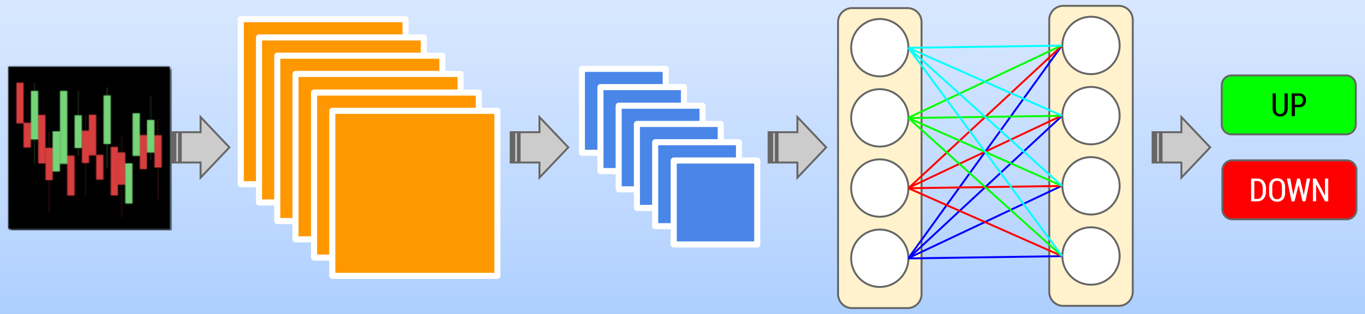
Modern Neural Network



VGG Network

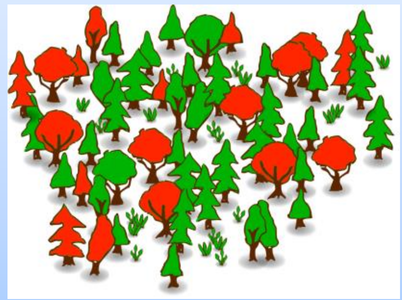


ResNet Network

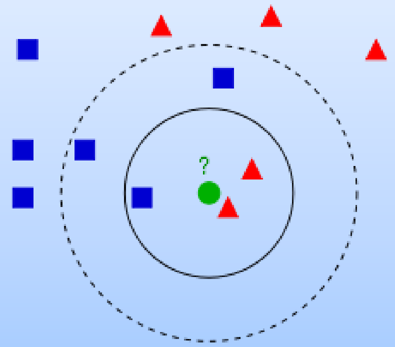


Convolutional Neural Network

Traditional Neural Network



Random Forest



K-Nearest Neighbor

Evaluation

Final Model

Top 10 - Taiwan50 Company List

No	Name	Ticker	Volume	52 Week Range
1	Advanced Semiconductor Engineering	2311.TW	86,190,484	44.15 - 46.10
2	AU Optronics	2409.TW	43,954,252	11.60 - 14.45
3	Hon Hai Precision Industry	2317.TW	36,224,415	79.50 - 122.50
4	China Development Financial Holdings	2883.TW	35,958,075	8.71 - 11.70
5	Taiwan Semiconductor Manufacturing	2330.TW	29,716,311	210.00 - 270.50
6	Siliconware Precision Industries	2325.TW	28,090,566	50.90 - 51.10
7	Innolux	3481.TW	27,146,129	10.80 - 16.30
8	E.Sun Financial Holding	2884.TW	20,970,569	17.70 - 21.9
9	United Microelectronics	2303.TW	20,428,290	13.40 - 18.65
10	Taiwan Cement	1101.TW	19,271,854	33.35 - 47.30

Full list : http://bit.ly/TAIWAN50COMPANIES

Indonesia 10 Company List

No	Name	Ticker	Volume	52 Week Range
1	Perusahaan Perseroan (Persero) PT Telekomunikasi Indonesia Tbk	TLKM.JK	120,850,800	3,250.00 - 4,840.00
2	PT Bank Rakyat Indonesia (Persero) Tbk	BBRI.JK	101,906,100	2,720.00 - 3,920.00
3	PT Bank Mandiri (Persero) Tbk	BMRI.JK	35,536,800	6,250.00 - 9,050.00
4	PT Bank Rakyat Indonesia (Persero) Tbk	ASII.JK	27,647,800	6,250.00 - 8,850.00
5	PT Bank Negara Indonesia (Persero) Tbk	BBNI.JK	25,450,600	6,750.00 - 10,175.00
6	PT Bank Central Asia Tbk	BBCA.JK	14,787,200	18,100.00 - 24,700.00
7	PT Bank Central Asia Tbk	HMSP.JK	12,466,700	3,230.00 - 5,550.00
8	PT United Tractors Tbk	UNTR.JK	4,970,000	27,625.00 - 40,500.00
9	PT Unilever Indonesia Tbk	UNVR.JK	1,317,800	43,875.00 - 58,100.00
10	PT Gudang Garam Tbk	GGRM.JK	413,900	61,925.00 - 86,400.00

From Historical Trading Data To Candlestick Chart

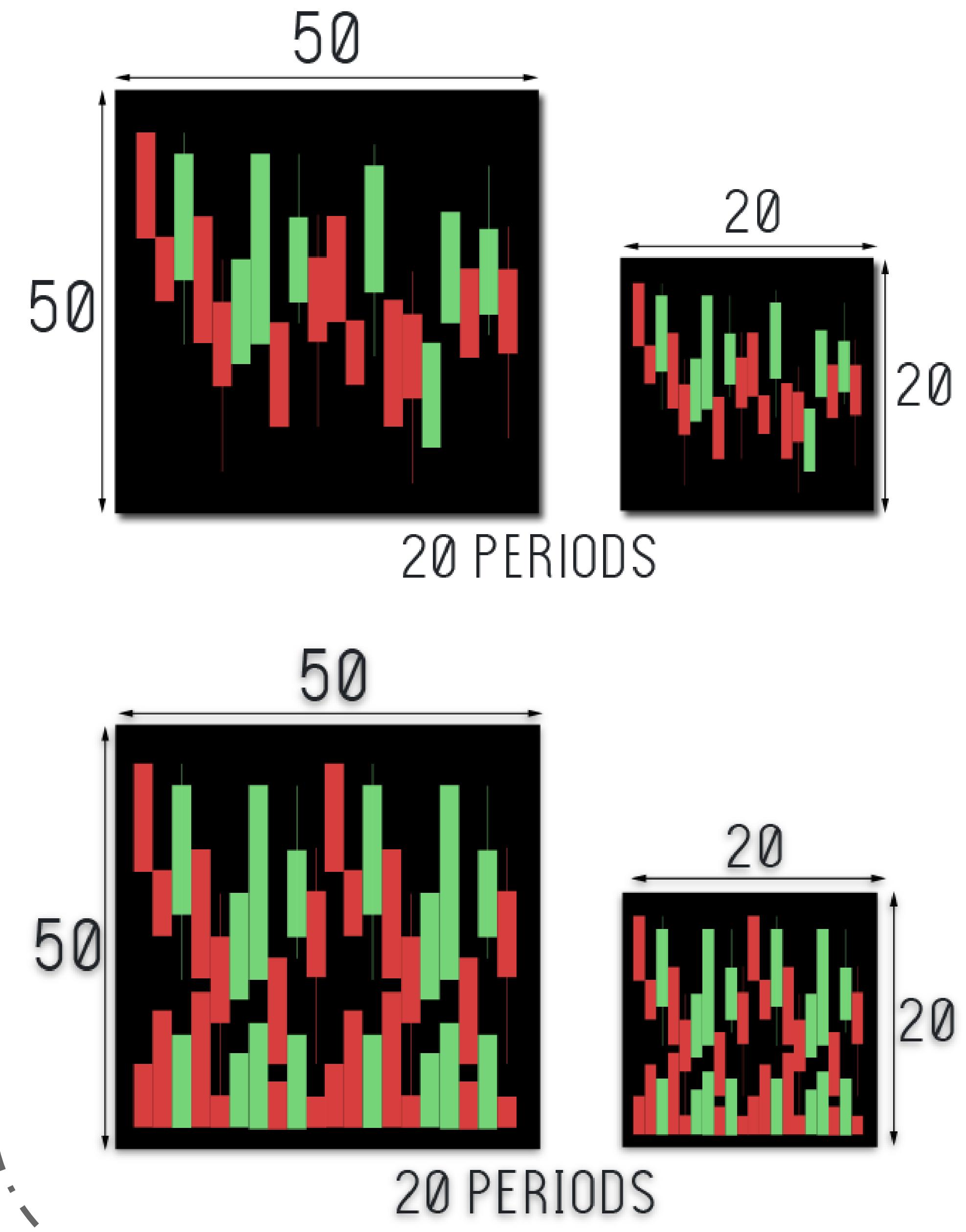
Date	Open	High	Low	Close*	Adj Close**	Volume
Jul 13, 2018	0.7700	0.8115	0.7700	0.8045	0.8045	14,205,544
Jul 12, 2018	0.8450	0.8450	0.7615	0.7800	0.7800	30,014,354
Jul 11, 2018	0.9000	0.9090	0.8360	0.8515	0.8515	42,052,086
Jul 10, 2018	0.8360	0.9030	0.8300	0.8980	0.8980	26,332,175
Jul 09, 2018	0.9120	0.9240	0.8220	0.8490	0.8490	39,492,461
Jul 06, 2018	0.8735	0.8995	0.8385	0.8785	0.8785	60,624,956
Jul 05, 2018	0.7600	0.8275	0.7555	0.8200	0.8200	39,754,178
Jul 04, 2018	0.7100	0.7390	0.7035	0.7375	0.7375	13,399,596
Jul 03, 2018	0.6680	0.6900	0.6640	0.6875	0.6875	2,944,229
Jul 02, 2018	0.6680	0.6680	0.6530	0.6660	0.6660	750,548
Jun 29, 2018	0.6515	0.6710	0.6515	0.6610	0.6610	867,891

Name	Training	Testing	Independent
TW50	198866	14756	348
ID10	34458	3692	348

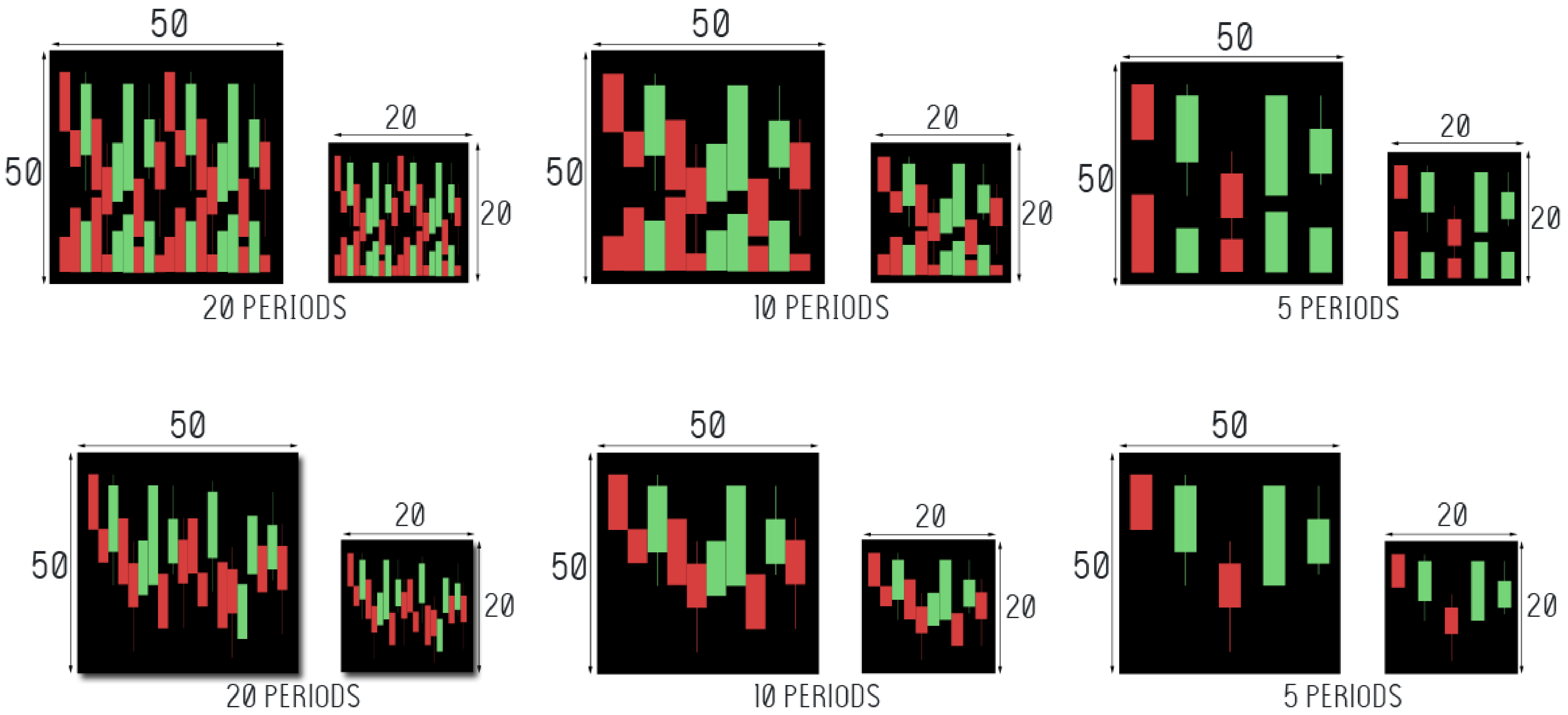
HISTORICAL TRADING DATA

matplotlib

CANDLESTICK CHART



Our Candlestick Chart



Sliding Window of Trading days Period

Our Dataset


Training and Testing Dataset

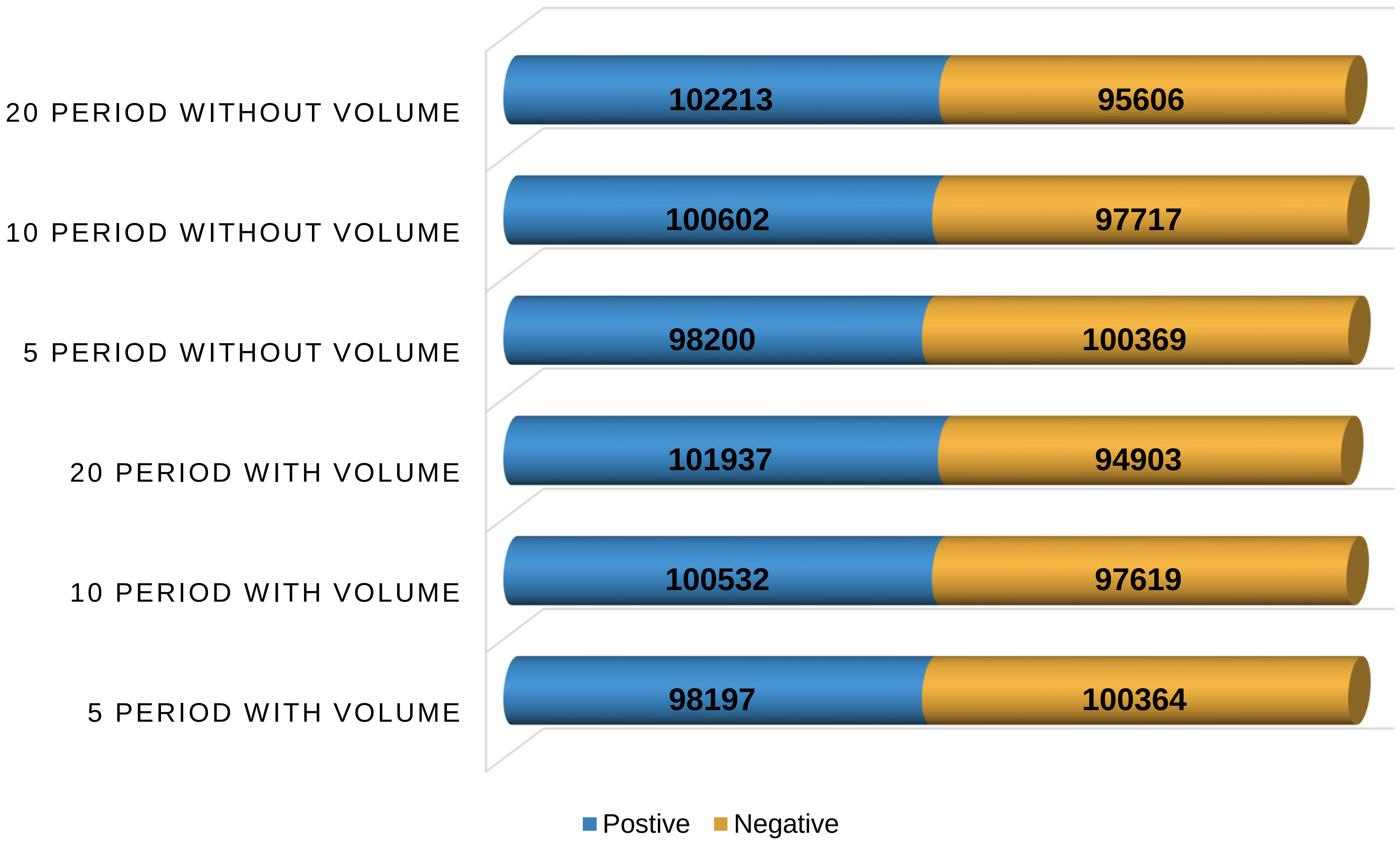
	5 Periods		10 Periods		20 Periods	
Name	Training	Testing	Training	Testing	Training	Testing
TW50	198569	17164	198151	16950	197819	16414
ID10	34350	3611	34323	3582	34233	3482

Independent Dataset

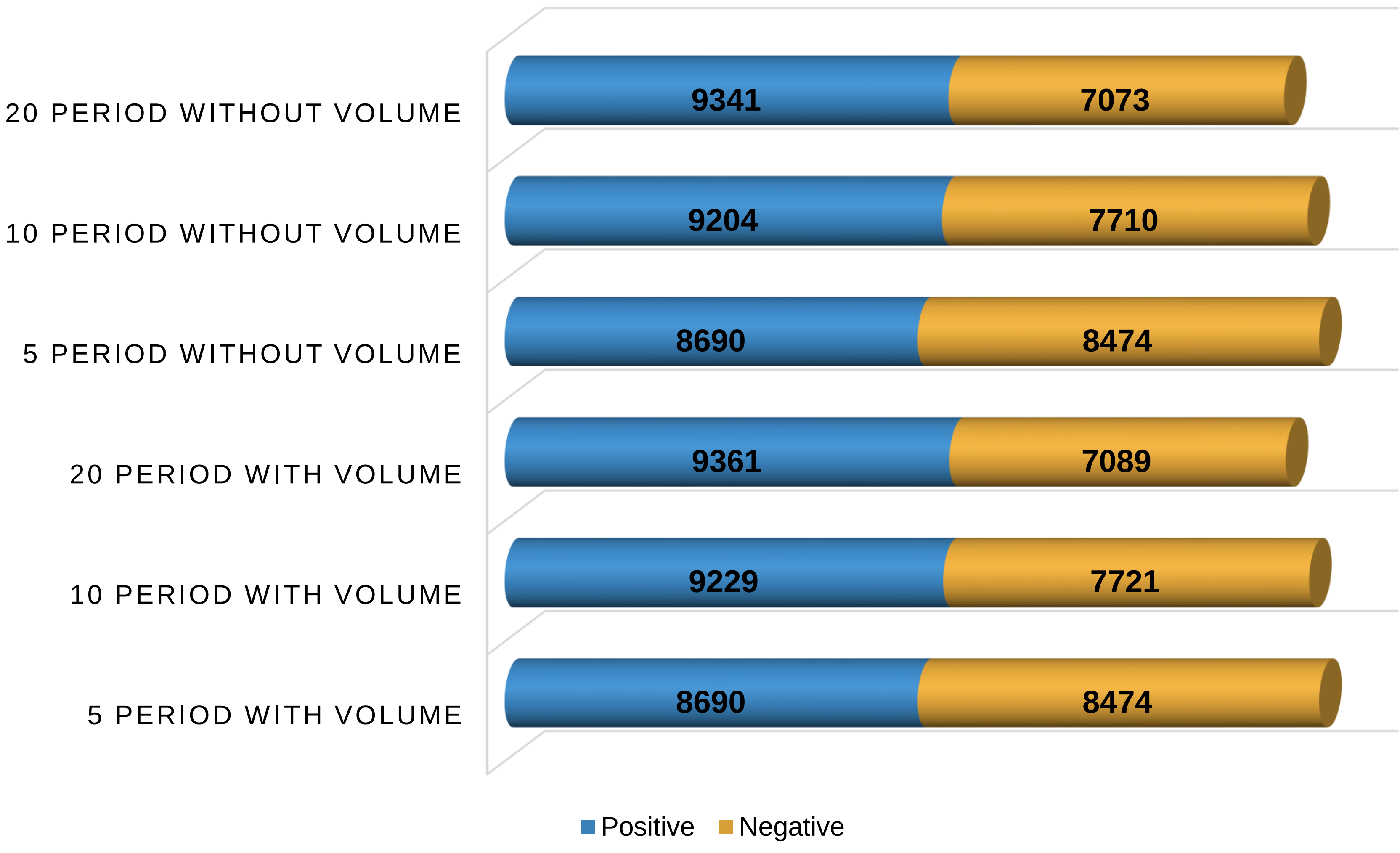
Data	5 Periods	10 Periods	20 Periods
0050.TW	342	337	327
^JKSE	342	337	327

Positive And Negative Statistic - TW50

TAIWAN50 – TRAINING DATASET

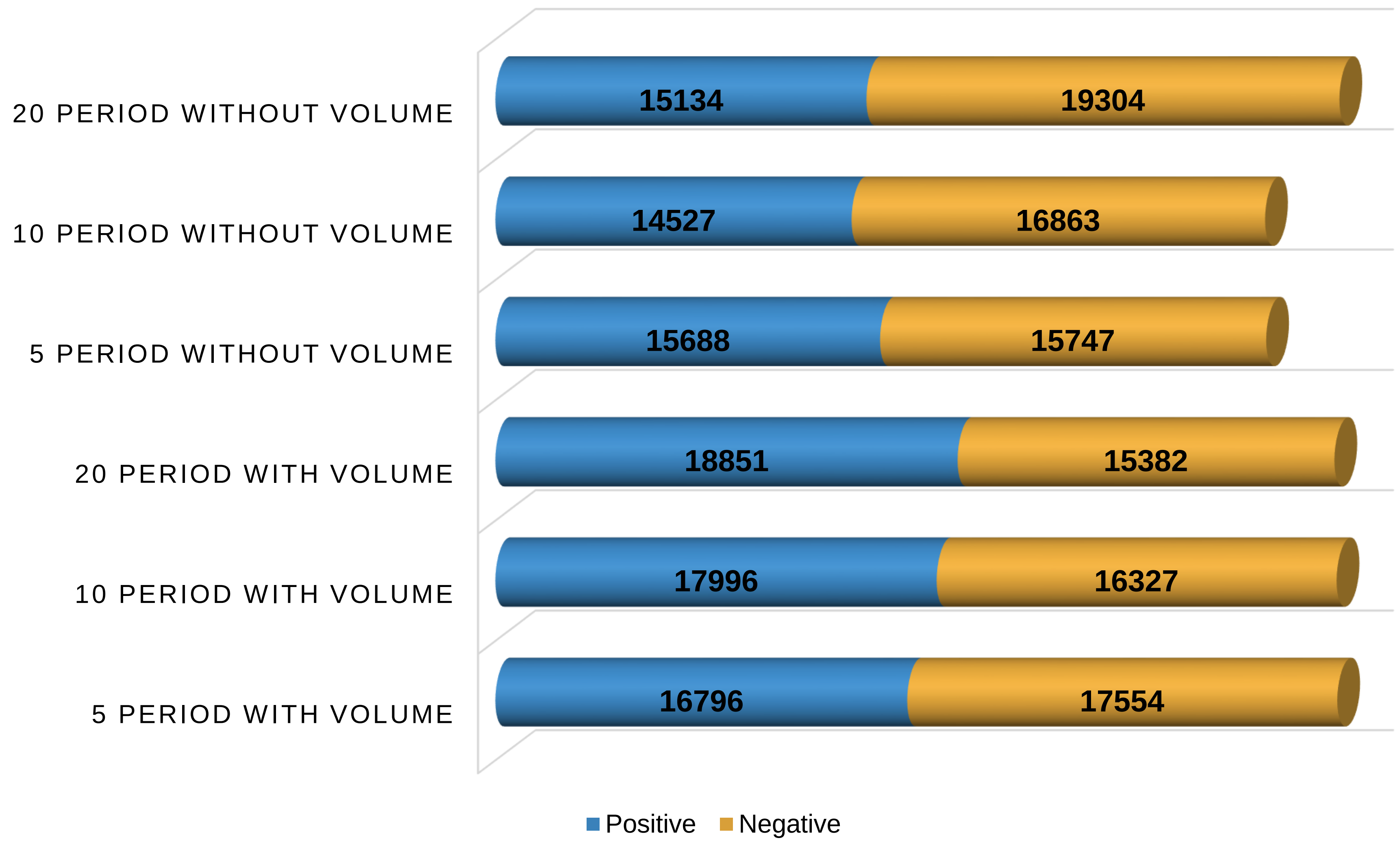


TAIWAN50 – TESTING DATASET

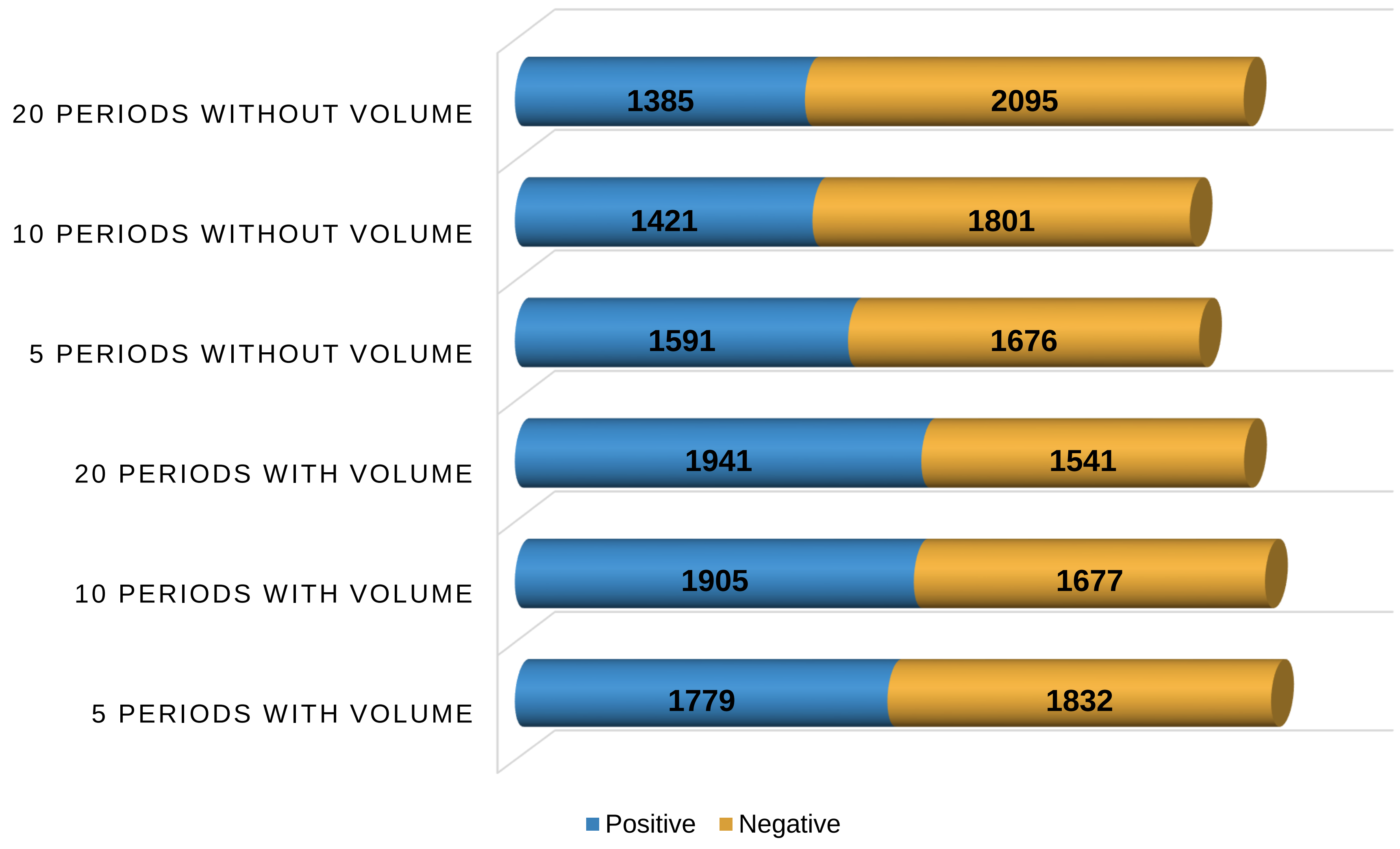


Positive And Negative Statistic - ID10

INDONESIA10 – TRAINING DATASET



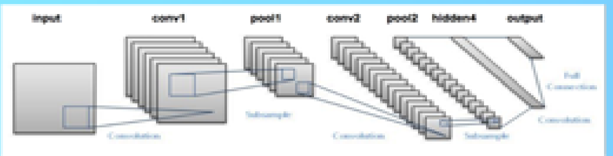
INDONESIA10 – TESTING DATASET



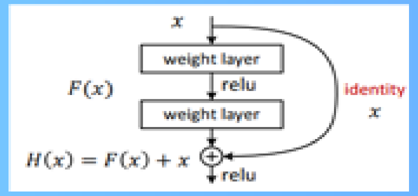
Learning Algorithm

MODERN NEURAL NETWORK

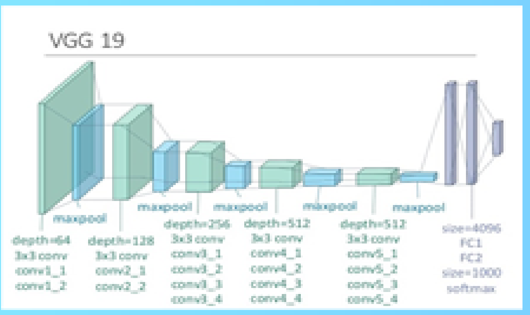
CNN



RESNET50



VGG19

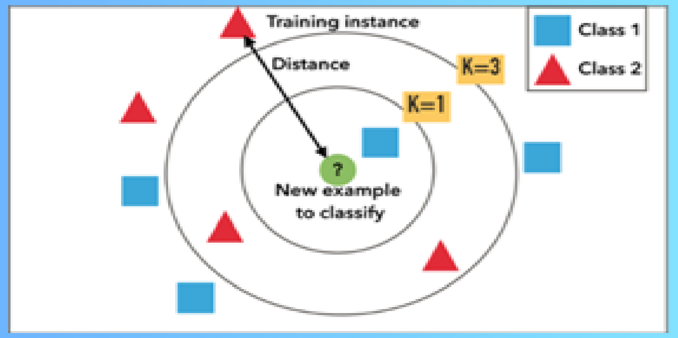


VGG16

TRADITIONAL NEURAL NETWORK

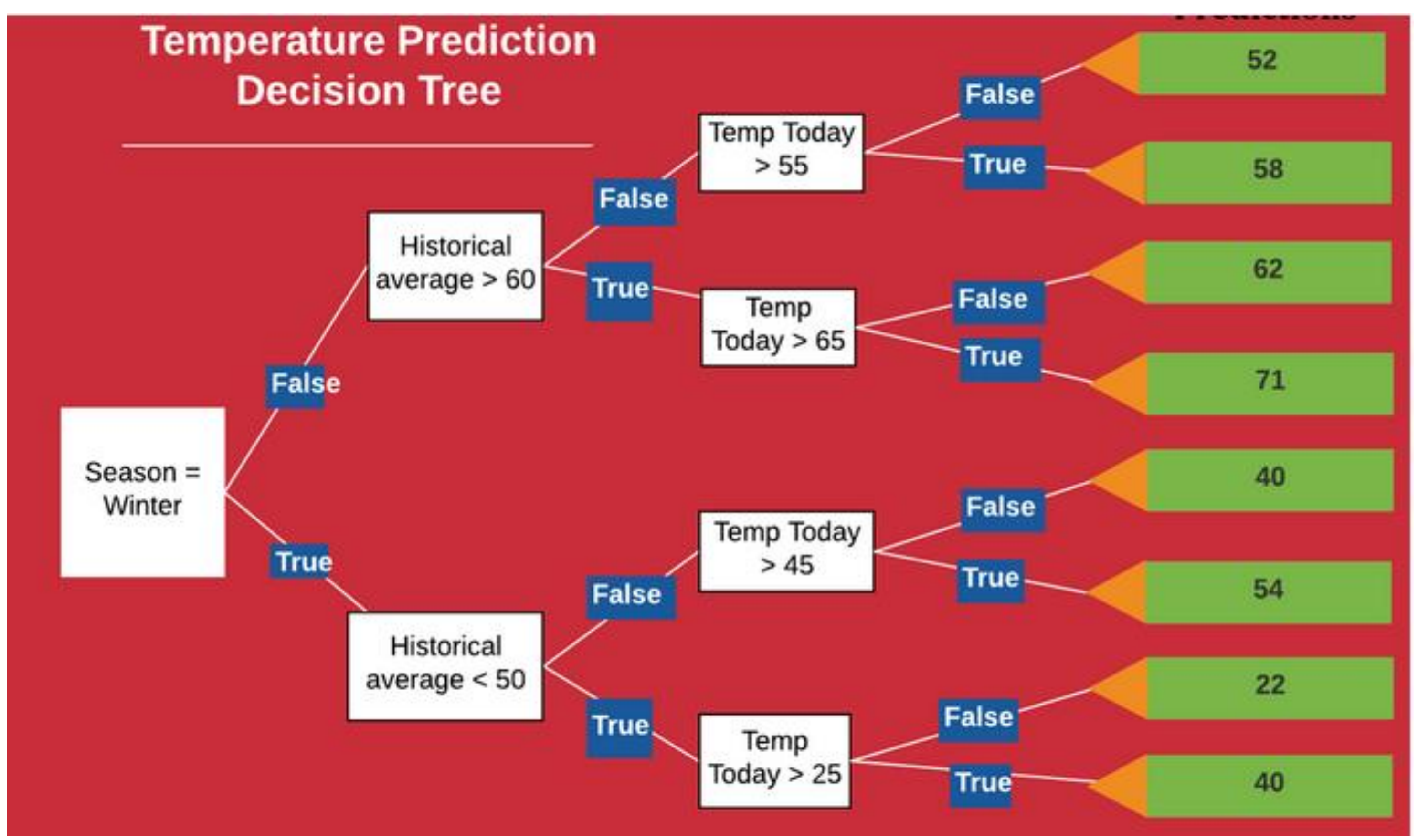
RANDOM FOREST

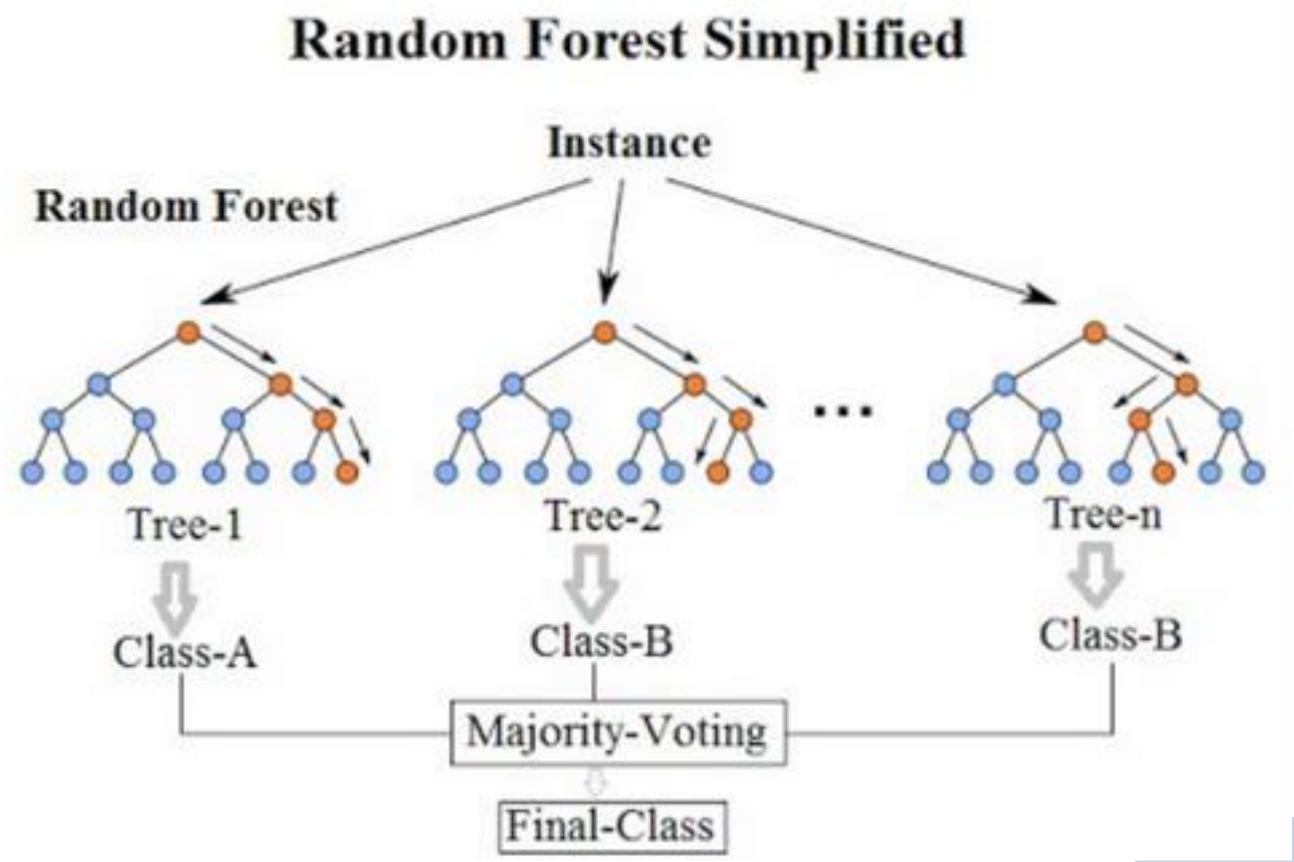
K-NN



Random Forest

- ❖ Decision trees use a **tree-like model** of decisions and their **possible consequences**.
- ❖ Random Forest classifier is a classifier with Consist of many decision trees. **Each decision tree** in the forest considers a **random subset** of features

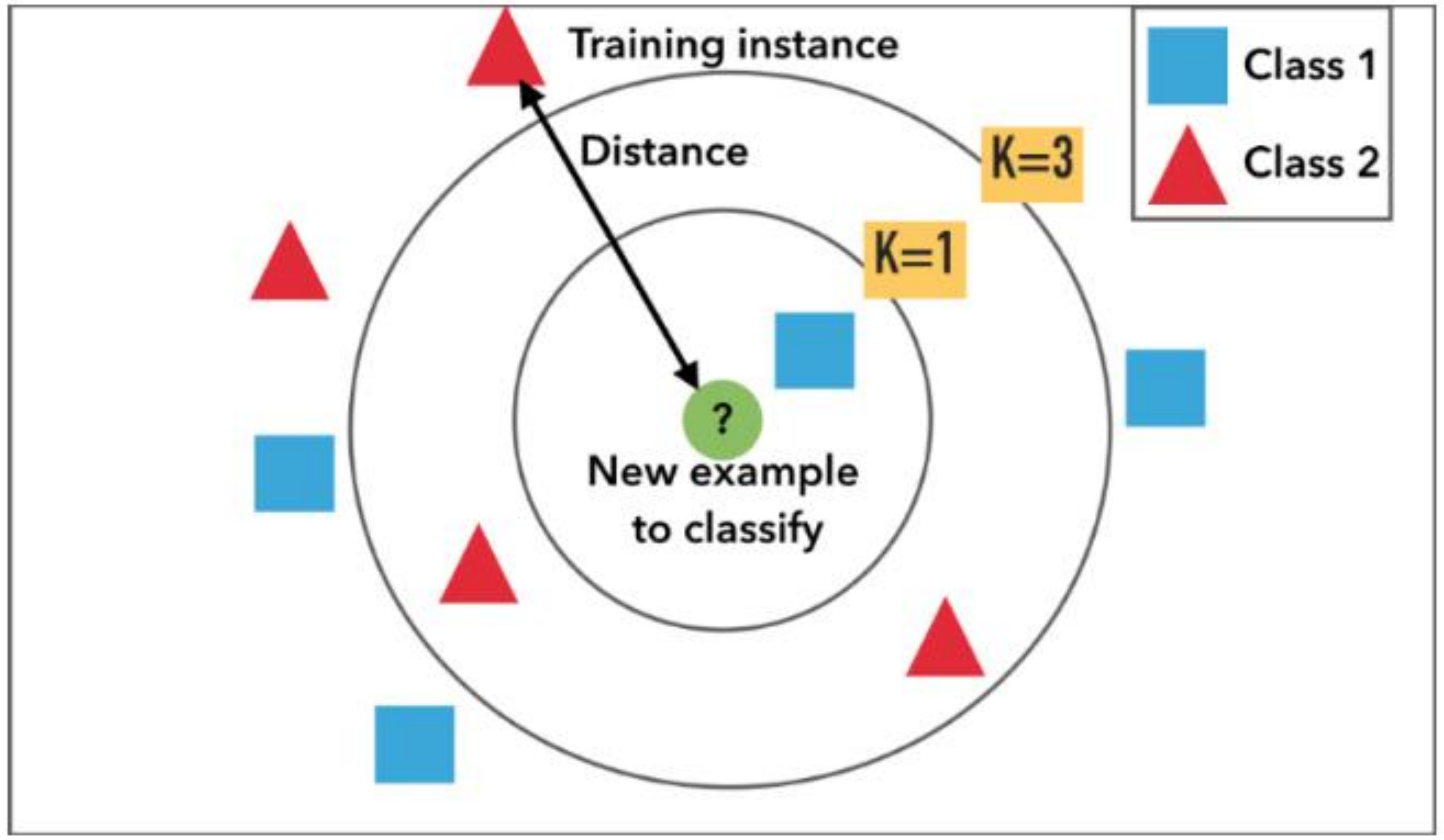




https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d

K-Nearest Neighbors

- A **non-parametric** method ,lazy learning algorithm that **categorizes** an input by using **its *k* nearest neighbors**
- **Separate** the data **points** into **several classes** to predict the **classification** of a **new point**
- **Determining a neighbor** can be performed using many different **notions of distance**, with the most common being **Euclidean** and **Hamming distance**



Euclidean distance $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Harming distance $D_H = \sum_{i=1}^{k}|x_i - y_i|$

https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7

Convolution Neural Networks

❖ ***Neural Network***:

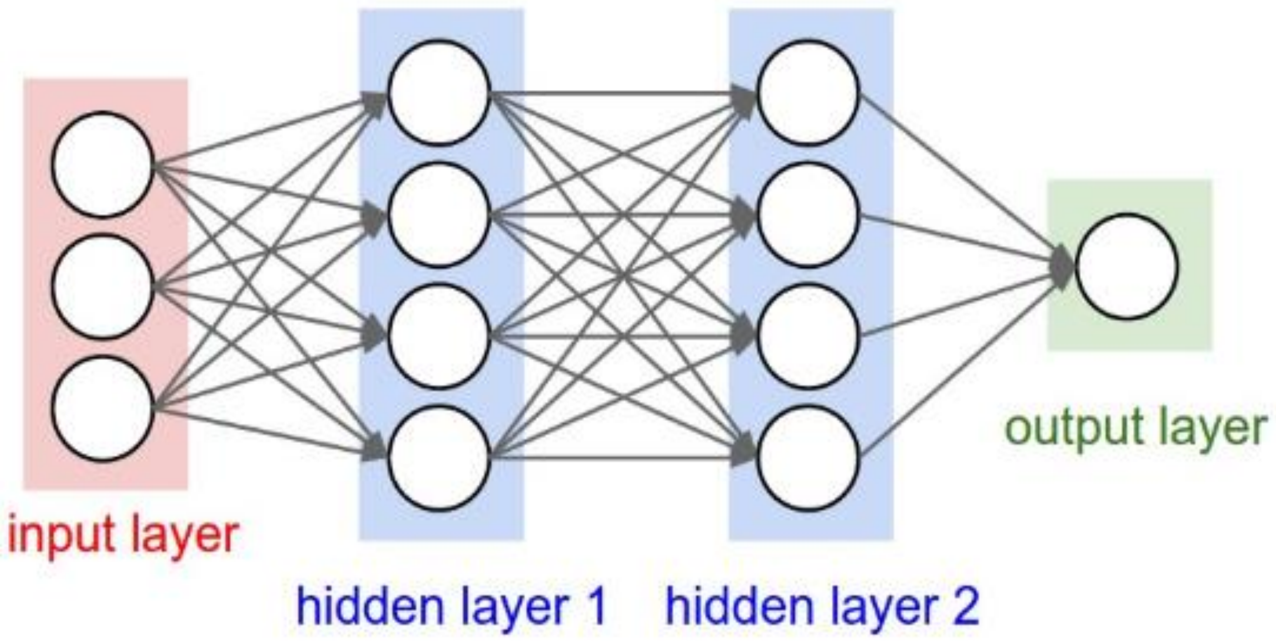
➢ **Modeled** as **collections of neurons** that are **connected** in an **acyclic graph**

➢ **outputs** of some **neurons** can become **inputs** to **other neurons**

➢ Receive an input, and transform it through a **series of hidden layers**.

➢ Each neuron is **fully connected** to **all neurons** in **previous layer**.

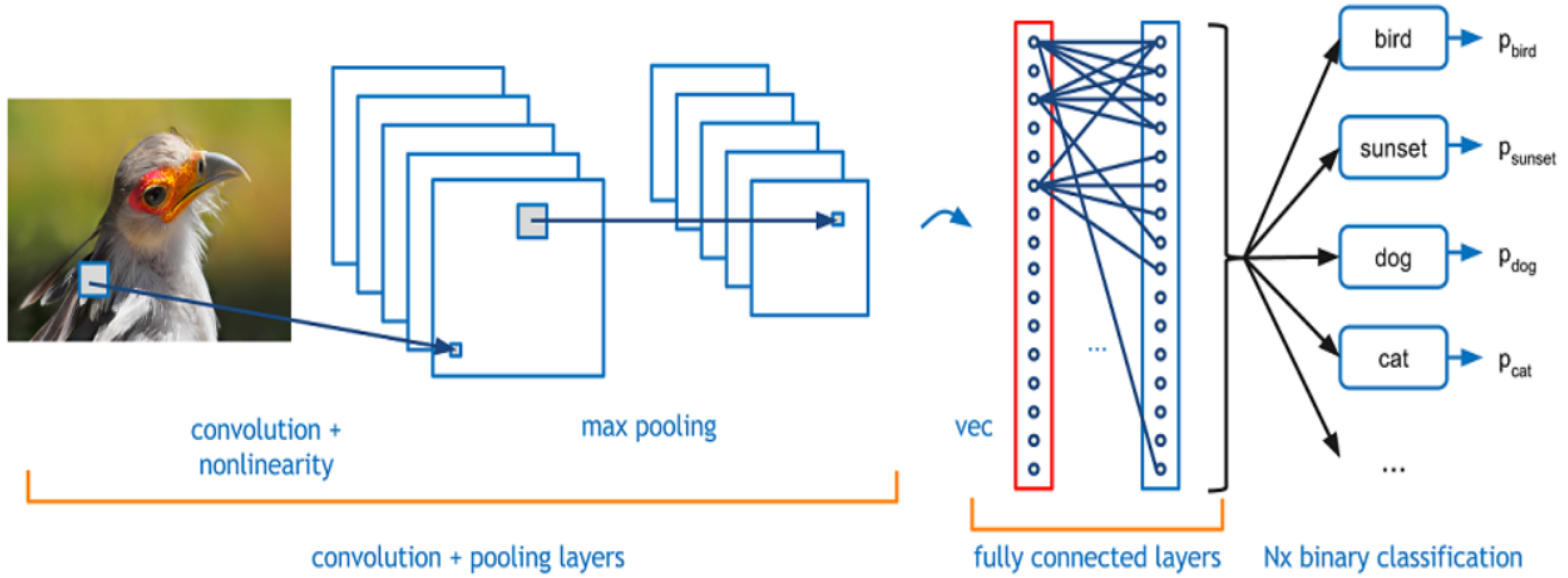
➢ Last **full-connected layer** is called the "**output layer**" (represents the **class scores** in classification task)



https://www.pyimagesearch.com/wp-content/uploads/2016/08/simple_neural_network_header-768x377.jpg

Convolution Neural Networks

- ***Convolution Neural Networks (ConvNets)***
  - **A** class of deep, feed-forward artificial neural networks.
  - **Neurons** that have **learnable weights** and **biases**
  - The hidden layers: **convolutional layers**, **pooling layers**, **fully connected layers** and **normalization layers**.
  - The **features** are **learned directly** by the CNN.
  - CNNs **can be retrained** for new recognition tasks

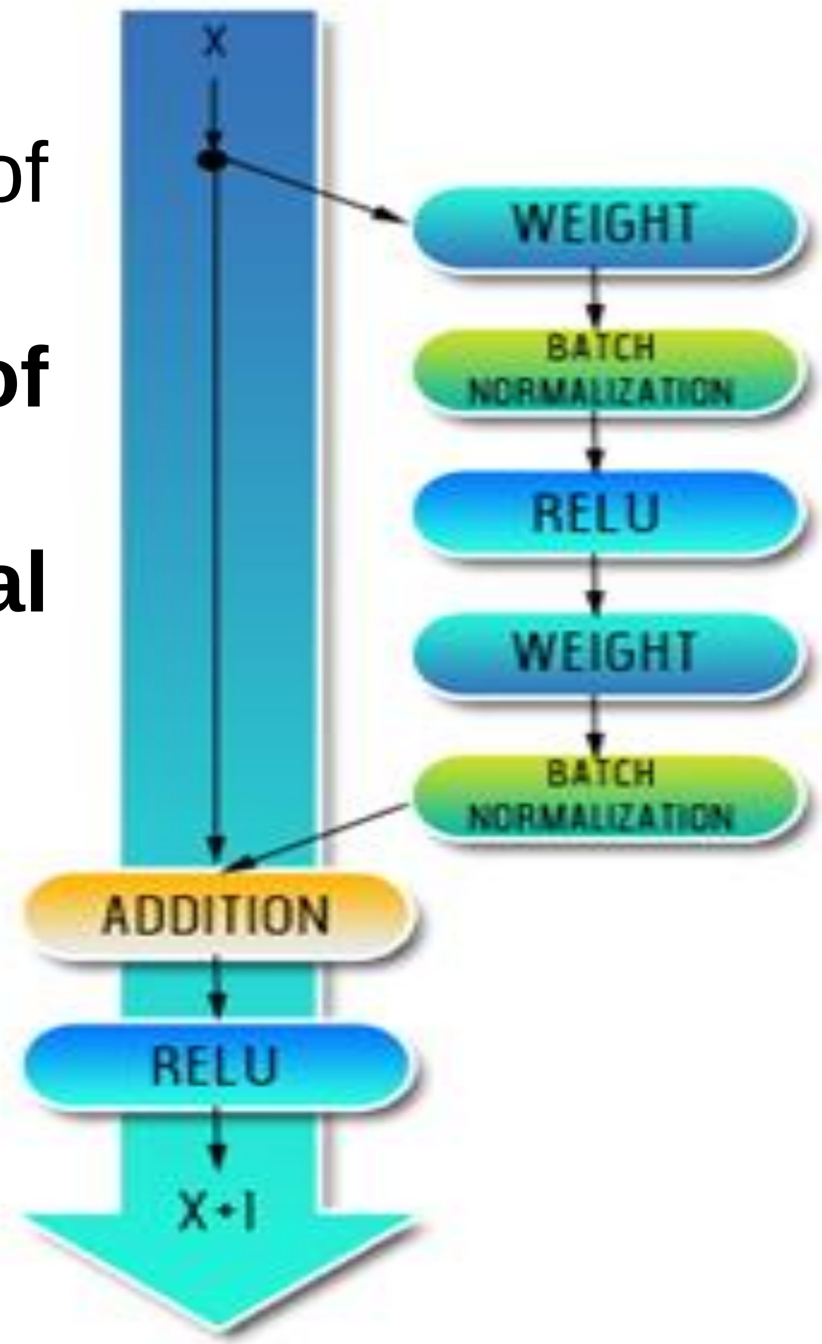


https://adeshpande3.github.io/assets/Cover.png

CNN Configuration
Input
Conv2D-32 ReLU
max-pooling
Conv2D-48 ReLU
max-pooling
Dropout
Conv2D-64 ReLU
max-pooling
Conv2D-96 ReLU
max-pooling
Dropout
Flatten
Dense-256
Dropout
Dense-2

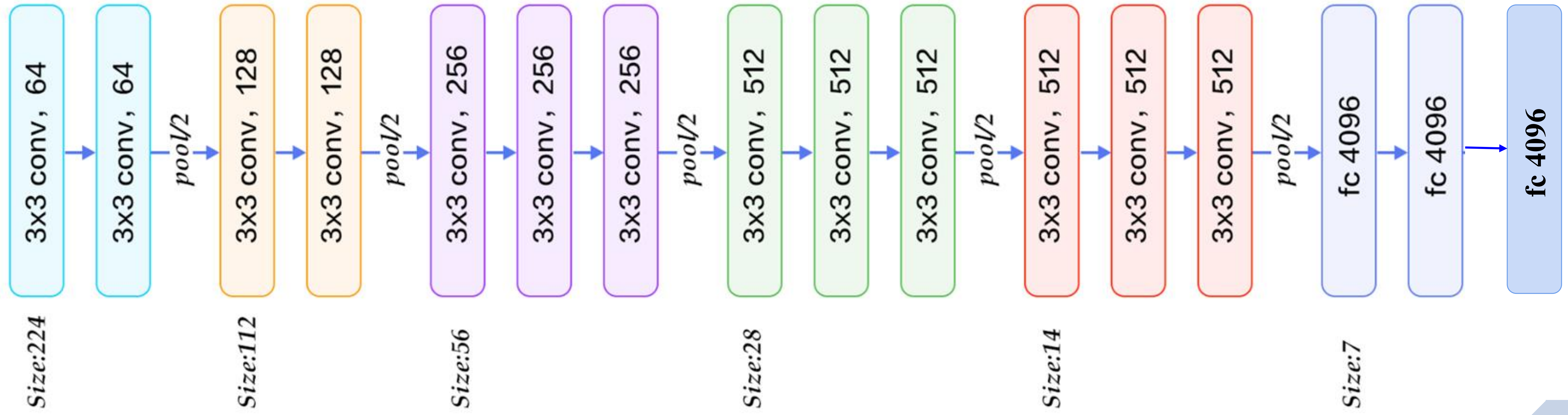
RESIDUAL NETWORK

- Developed by (He, Zhang et al. 2016) was the winner of ILSVRC 2015.
- It features special **skip connections** and a **heavy use of batch normalization**.
- ResNets are currently by **far state of the art Convolutional Neural Network models**



Visual Geometry Group Network

- The VGG network architecture was introduced by (Simonyan and Zisserman 2014).
- using only **3x3 convolutional layers stacked** on top of each other in increasing depth.
- **Reducing volume size** is handled by **max pooling**



https://www.quora.com/What-is-the-VGG-neural-network

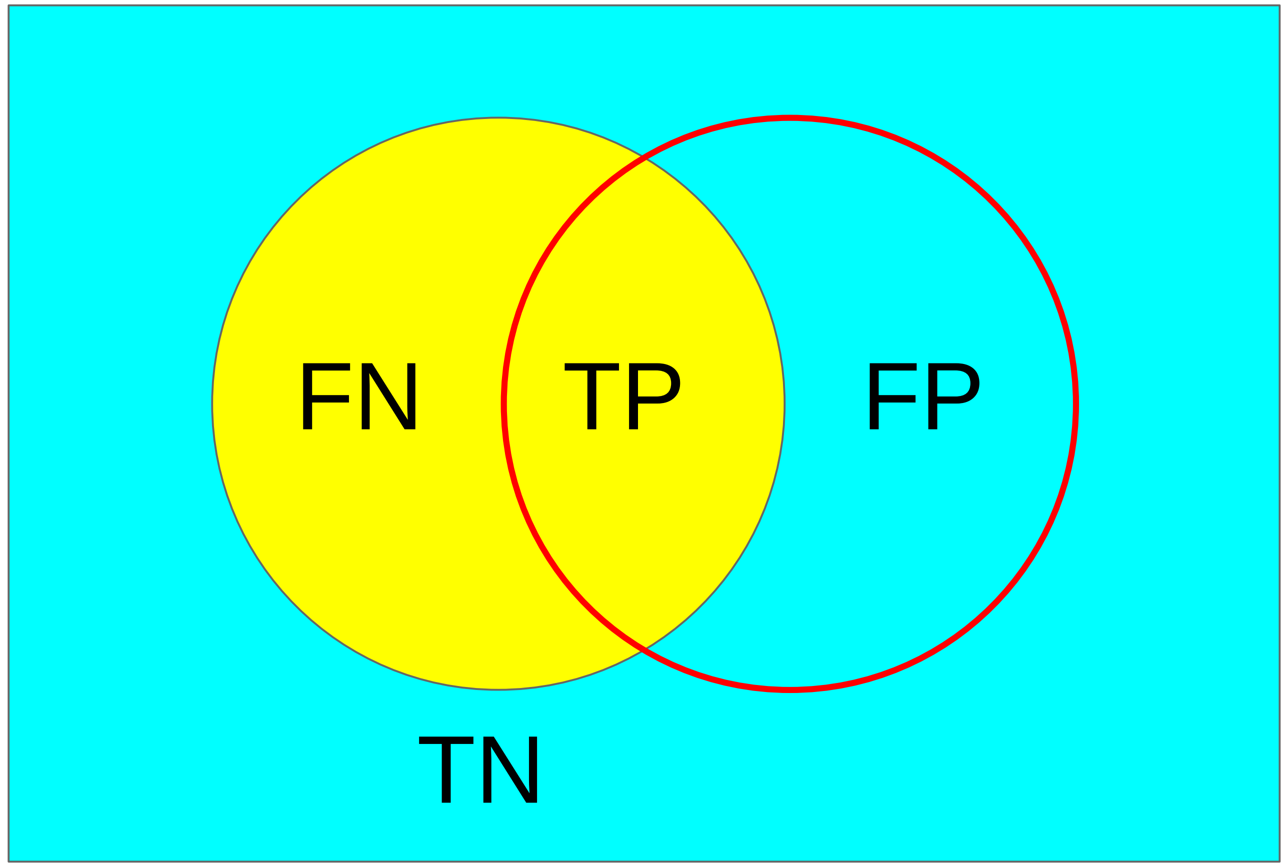
Performance Evaluation

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specitivity = \frac{TN}{TN + FP}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$MCC = \frac{TPxTN - FPxFN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$



TESTING RESULT

TW50 and ID10

Testing Result TW50

	Classifier	Periods	Dimension	Sensitivity	Specificity	Accuracy	MCC
With Volume	CNN	5	50	83.2	83.8	83.5	0.67
	CNN	10	50	88.6	87.3	88.0	0.758
	CNN	20	50	91.6	91.3	91.5	0.827
	CNN	5	20	83.9	82.7	83.3	0.666
	Random Forest	10	20	87.0	88.3	87.6	0.751
	CNN	20	20	90.8	90.2	90.6	0.808
Without Volume	CNN	5	50	83.6	85.1	84.4	0.687
	CNN	10	50	89.2	88.1	88.7	0.773
	CNN	**20**	**50**	**93.3**	**90.7**	**92.2**	**0.84**
	CNN	5	20	84.8	83.0	83.9	0.678
	CNN	10	20	88.0	88.2	88.1	0.761
	CNN	20	20	81.7	91.4	91.0	0.817

Testing Result ID10

	Classifier	Periods	Dimension	Sensitivity	Specificity	Accuracy	MCC
With Volume	ResNet50	5	50	80.7	85.4	83.1	0.661
	ResNet50	10	50	88.6	88.4	88.5	0.77
	CNN	20	50	90.0	90.1	90.0	0.798
	ResNet50	5	20	78.8	82.3	80.6	0.612
	CNN	10	20	83.3	85.4	84.3	0.686
	CNN	20	20	89.1	84.6	87.1	0.738
Without Volume	ResNet50	5	50	79.1	87.9	83.3	0.671
	CNN	10	50	87.5	86.6	87.1	0.74
	CNN	**20**	**50**	**92.1**	**92.1**	**92.1**	**0.837**
	CNN	5	20	83.4	82.4	82.9	0.658
	CNN	10	20	85.4	85.6	85.5	0.708
	VGG16	20	20	91.5	89.7	90.7	0.808

INDEPENDENT TEST RESULT

TW50 and ID10

Independent Result TW50

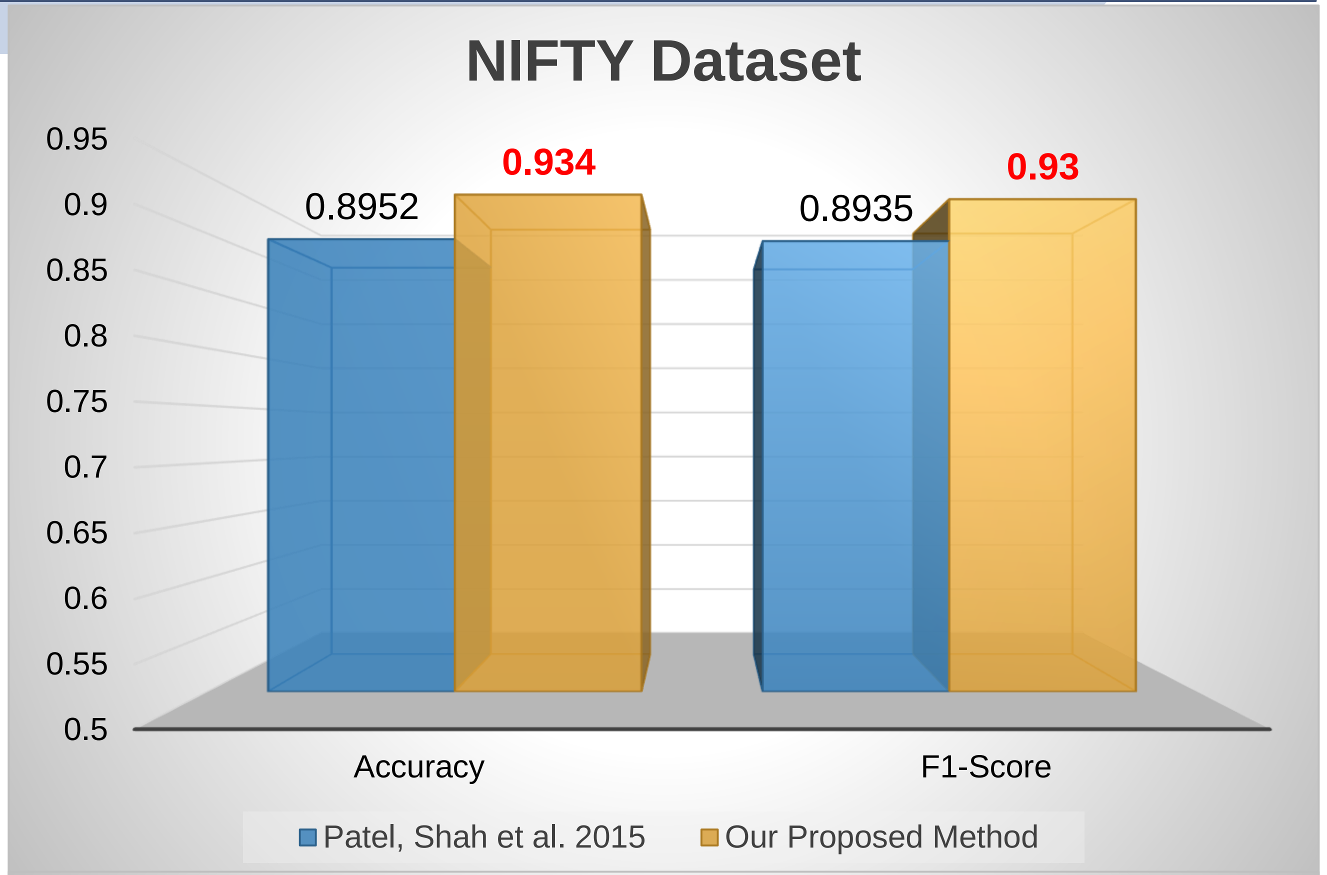
	Classifier	Periods	Dimension	Sensitivity	Specificity	Accuracy	MCC
With Volume	CNN	5	50	82.1	77.1	79.9	0.593
	CNN	10	50	85.7	81.5	84.0	0.669
	CNN	20	50	95.8	87.1	92.7	0.839
	CNN	5	20	82.6	83.0	82.8	0.654
	Random Forest	10	20	44.8	84.4	60.7	0.305
	CNN	20	20	92.9	89.7	91.8	0.821
Without Volume	CNN	5	50	82.1	81.0	81.6	0.63
	CNN	10	50	89.7	83.7	87.3	0.735
	CNN	**20**	**50**	**94.3**	**91.4**	**93.3**	**0.854**
	CNN	5	20	81.1	82.4	81.6	0.631
	CNN	10	20	89.7	86.7	88.5	0.761
	CNN	20	20	92.0	93.1	92.4	0.838

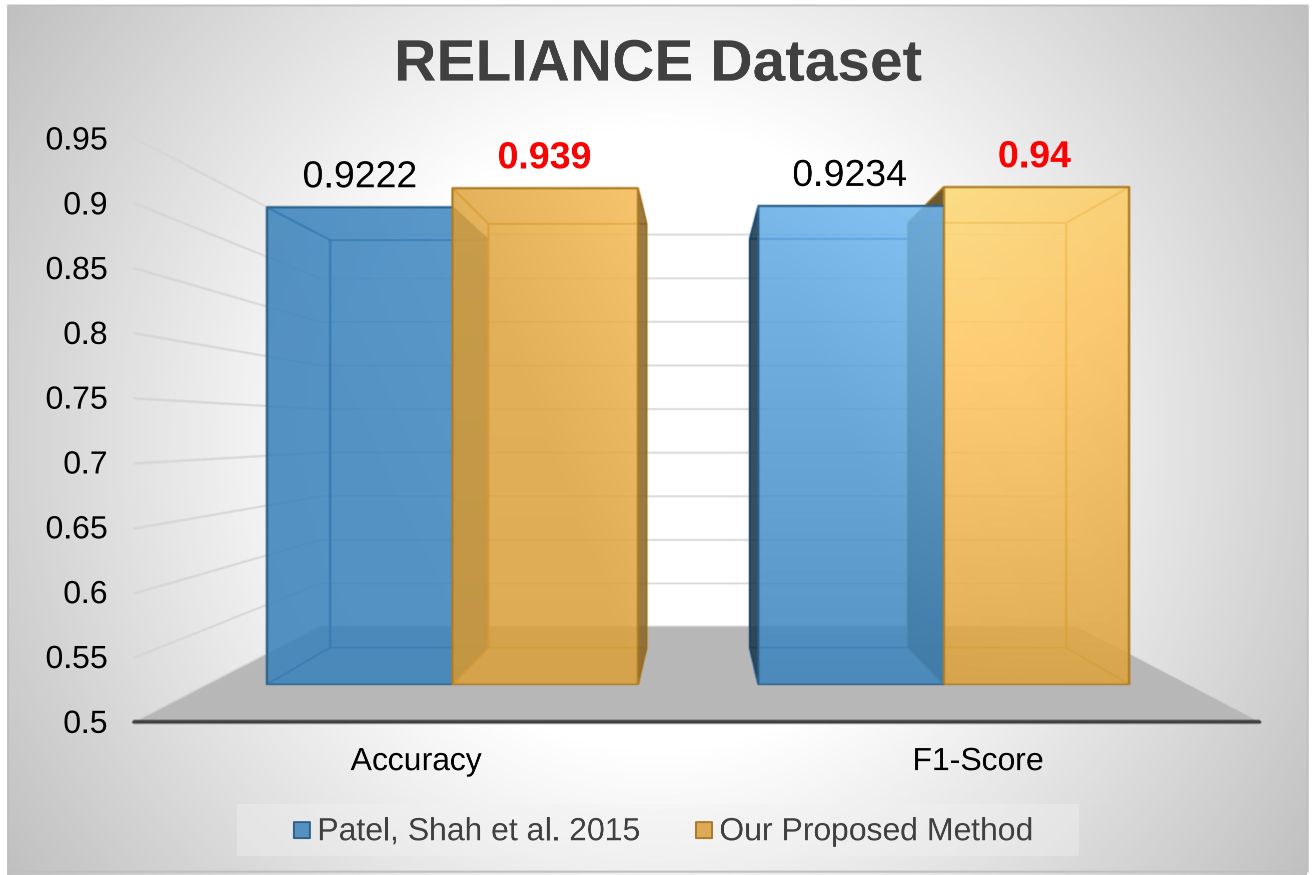
Independent Result ID10

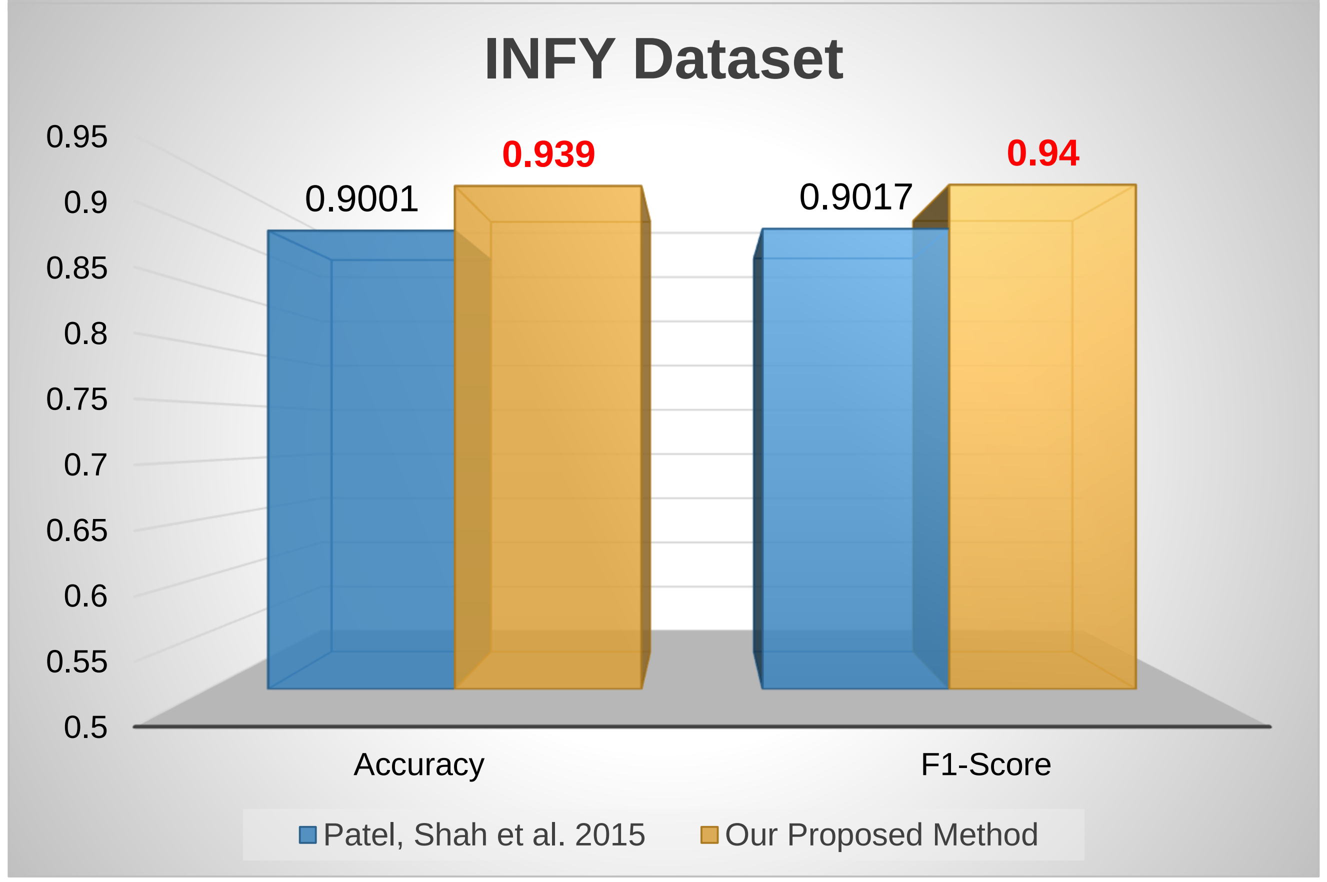
	Classifier	Periods	Dimension	Sensitivity	Specificity	Accuracy	MCC
With Volume	RESNET50	5	50	80.8	88.8	83.9	0.681
	RESNET50	10	50	90.9	84.7	89.3	0.733
	CNN	20	50	87.2	83.5	86.2	0.67
	RESNET50	5	20	75.0	82.1	77.8	0.558
	CNN	10	20	83.9	75.3	81.7	0.559
	CNN	20	20	83.9	82.4	83.5	0.616
Without Volume	RESNET50	5	50	79.3	86.6	82.2	0.645
	CNN	10	50	90.6	87.1	89.3	0.772
	CNN	**20**	**50**	**90.6**	**88.7**	**89.9**	**0.786**
	CNN	5	20	81.7	78.4	80.4	0.595
	CNN	10	20	86.9	88.7	87.5	0.741
	VGG16	20	20	91.3	81.2	88.7	0.712

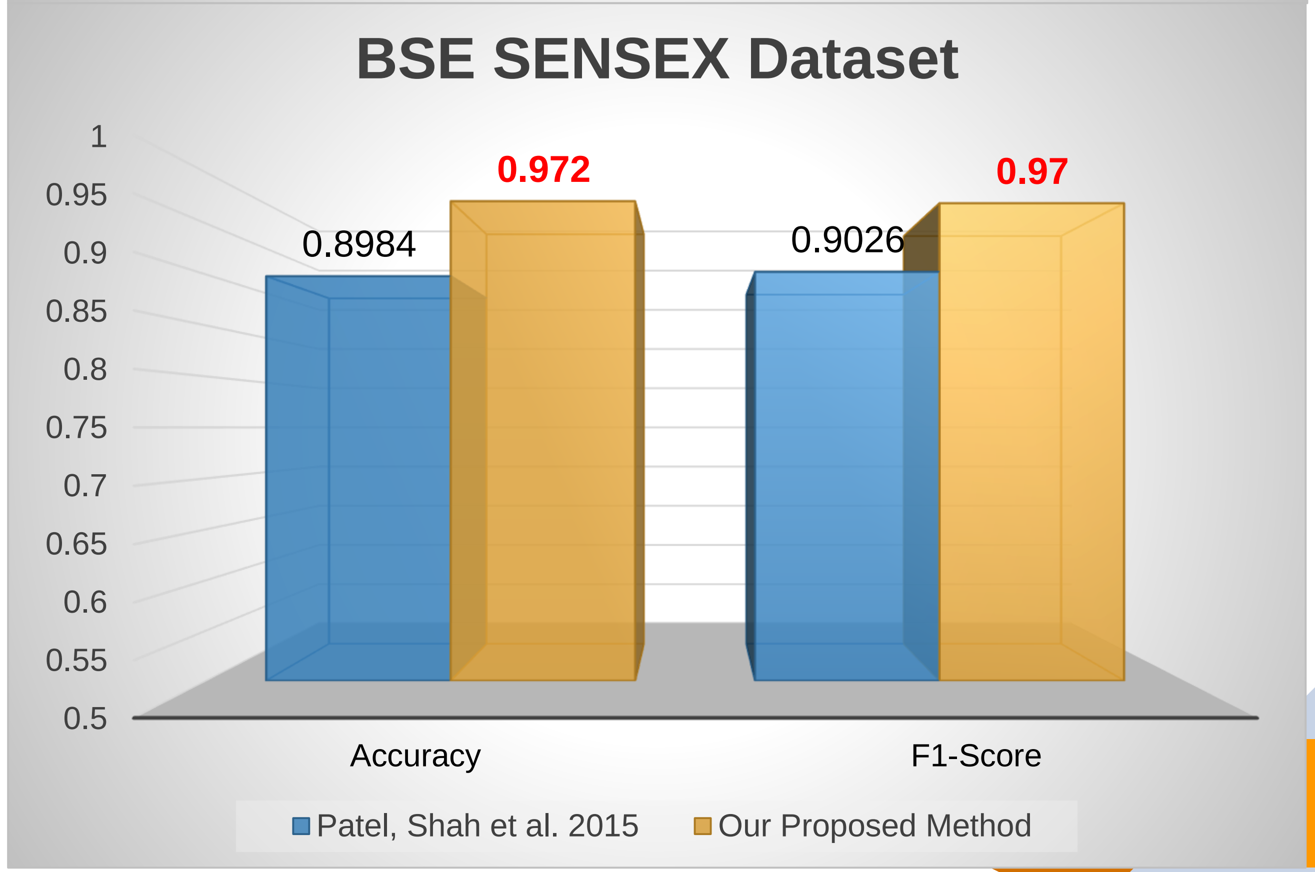
COMPARISON

Comparison - Patel, Shah et al. 2015



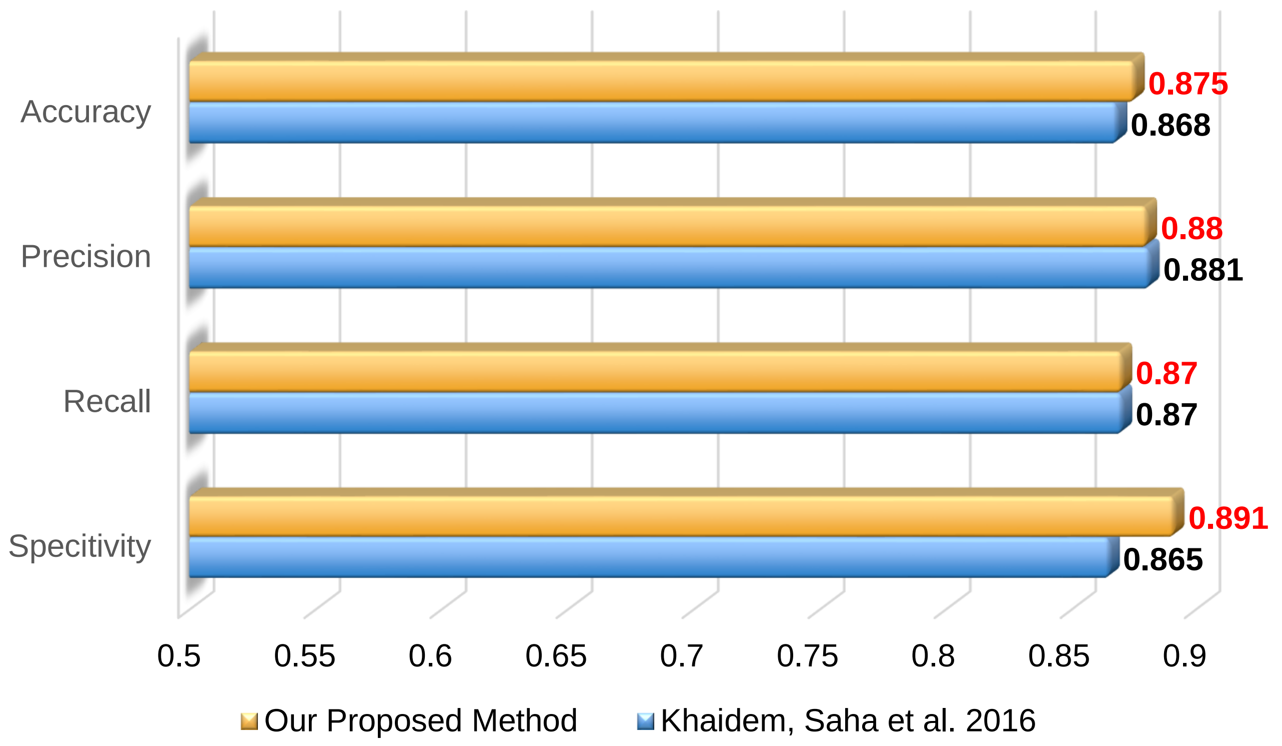




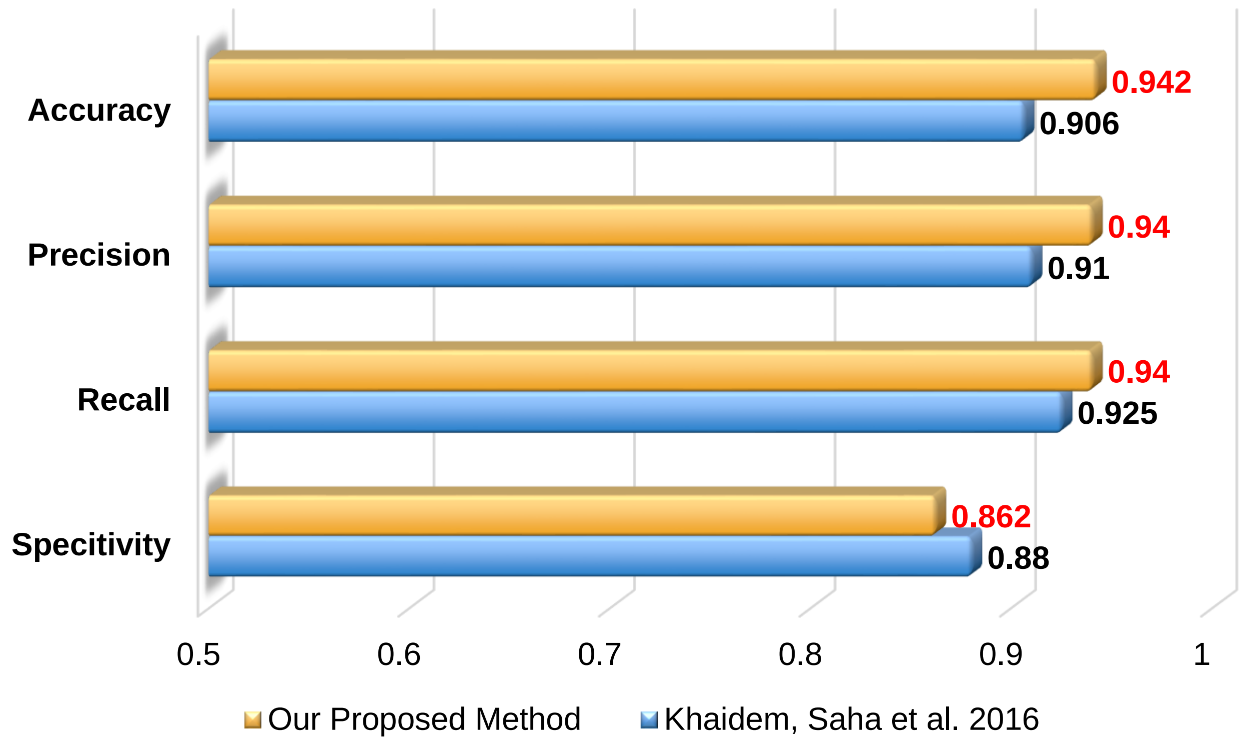


Comparison - Khaidem, Saha et al. 2016 - Samsung

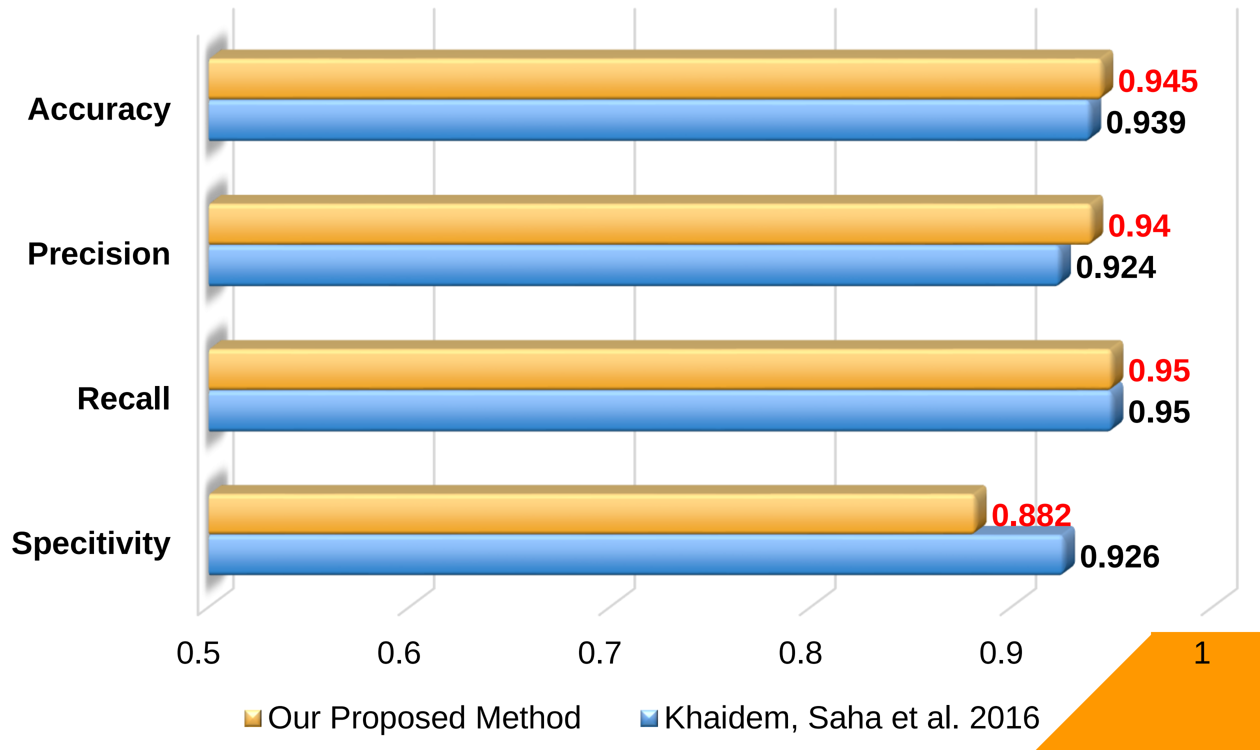
1 Month Periods



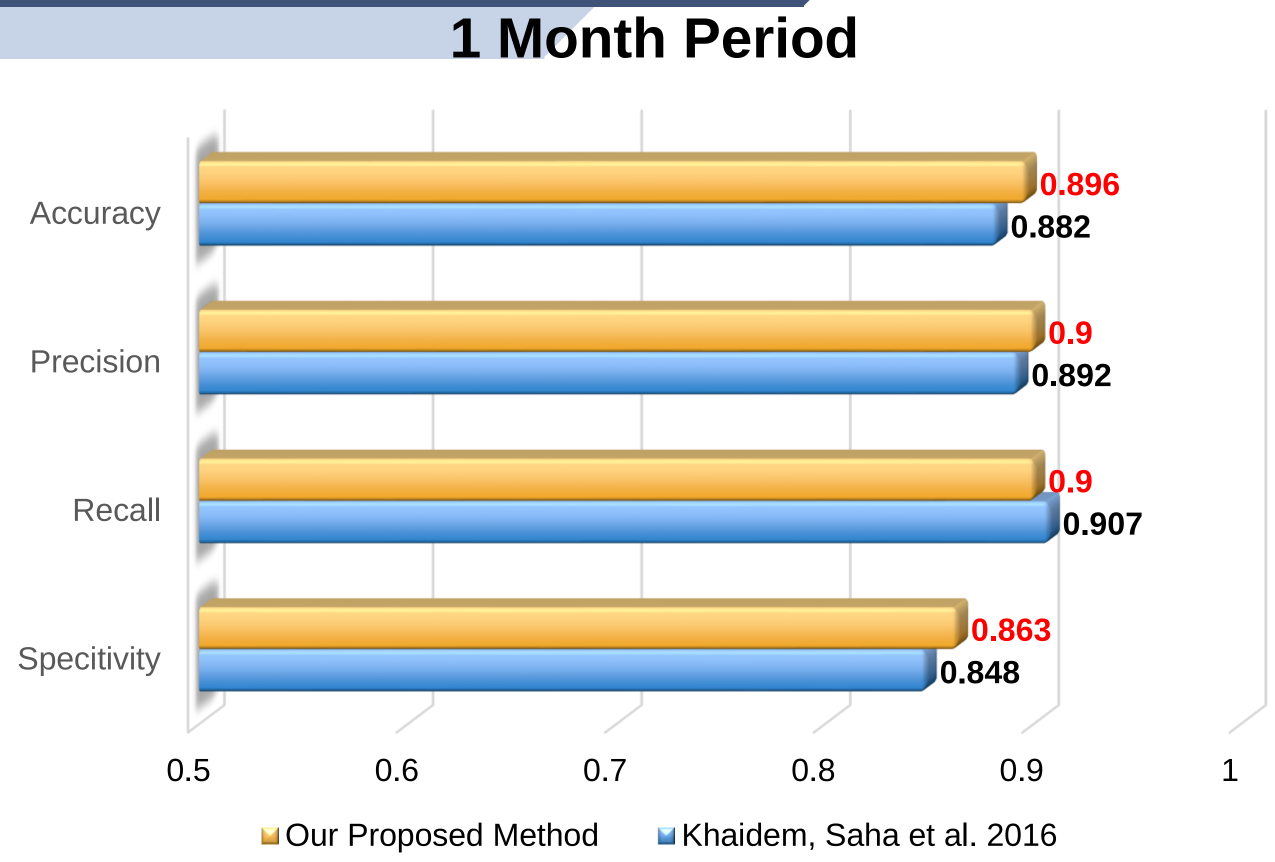
2 Months Period

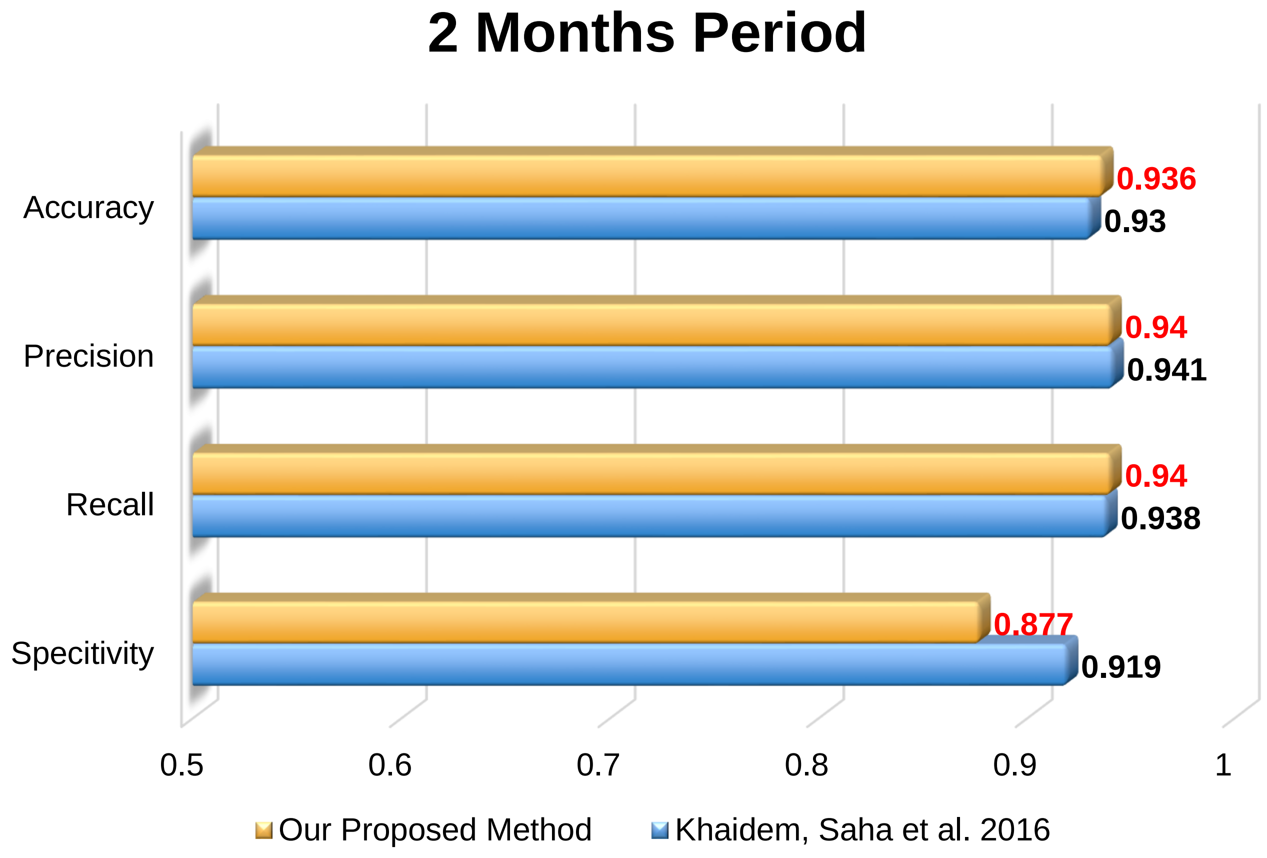


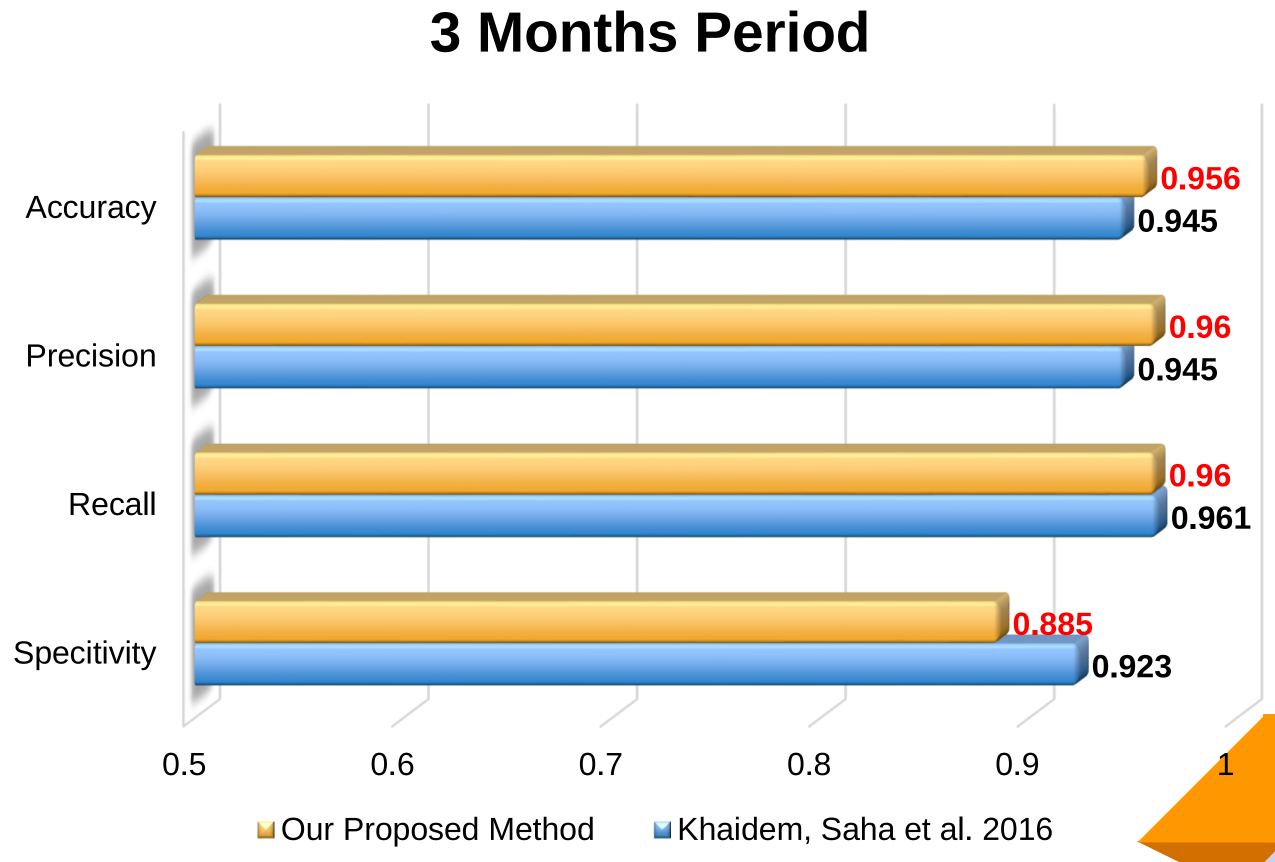
3 Months Period



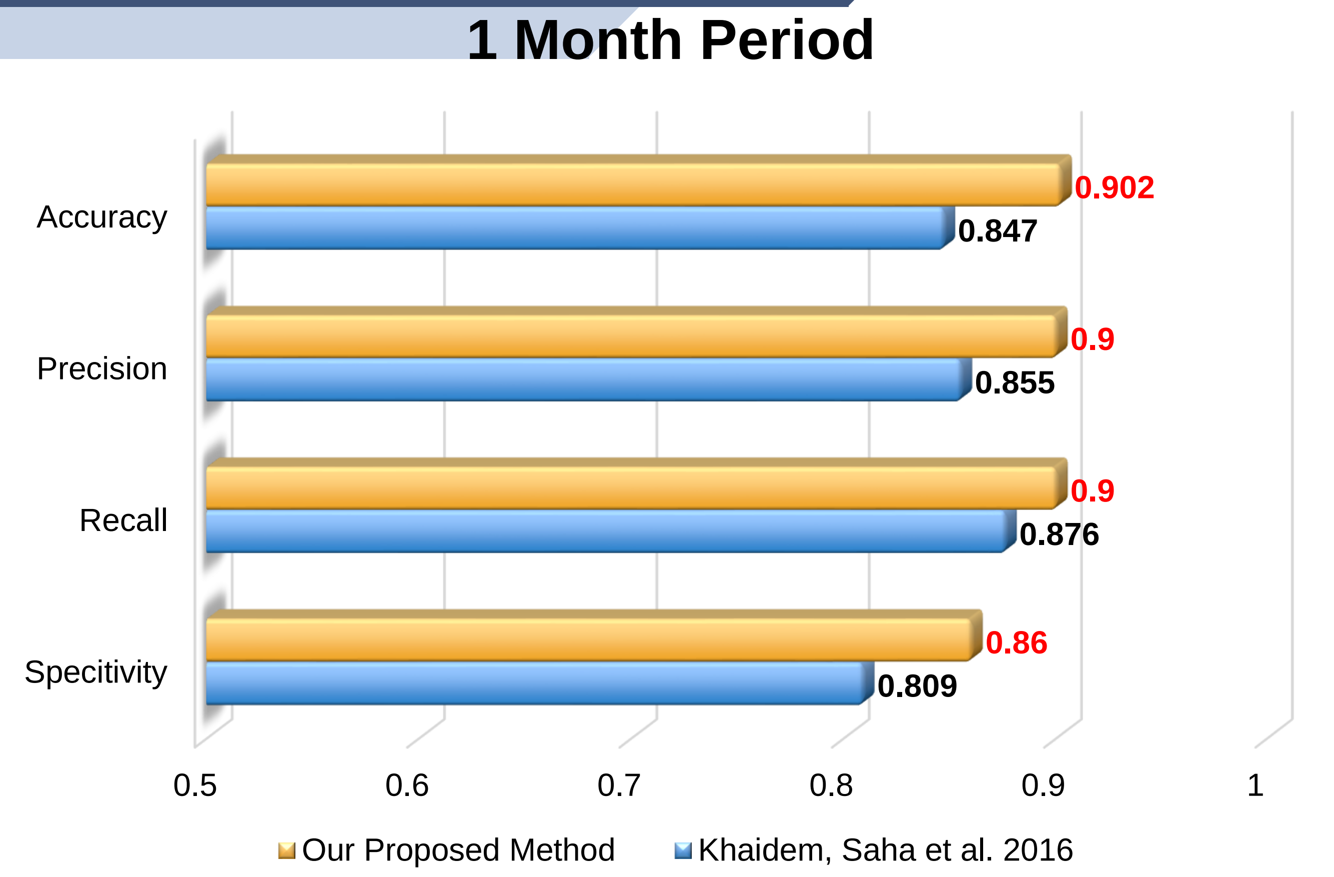
Comparison - Khaidem, Saha et al. 2016 - Apple

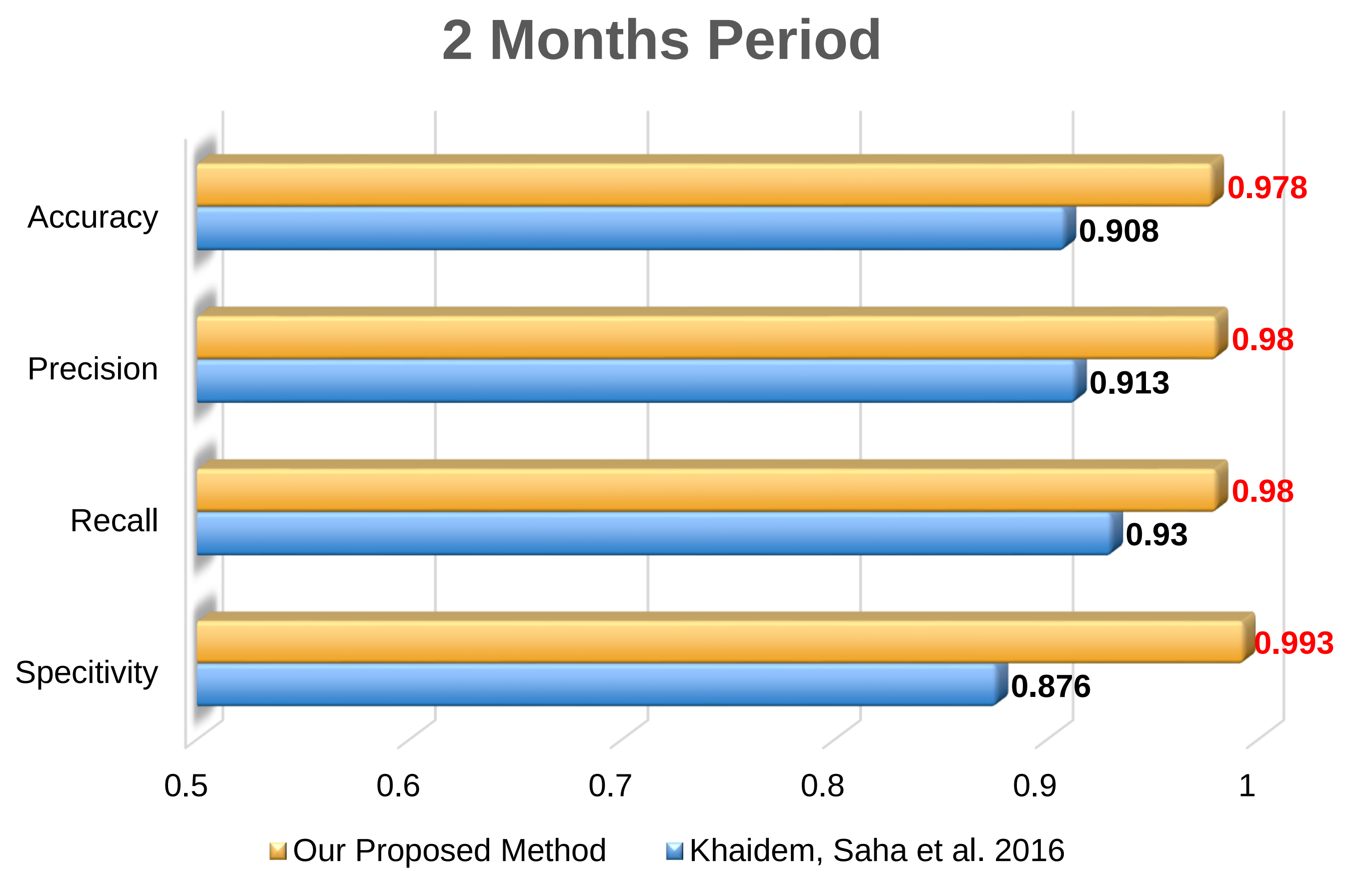


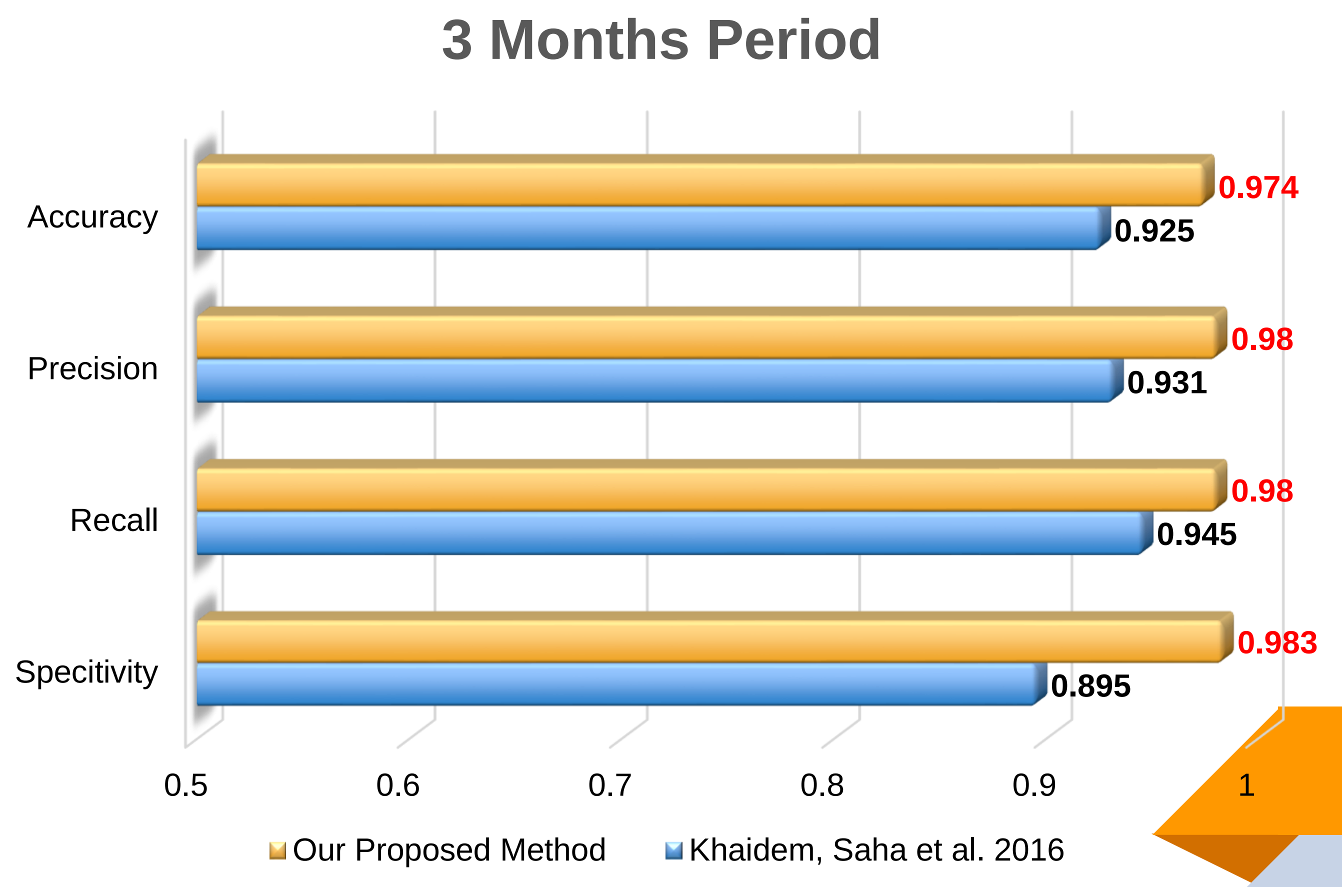




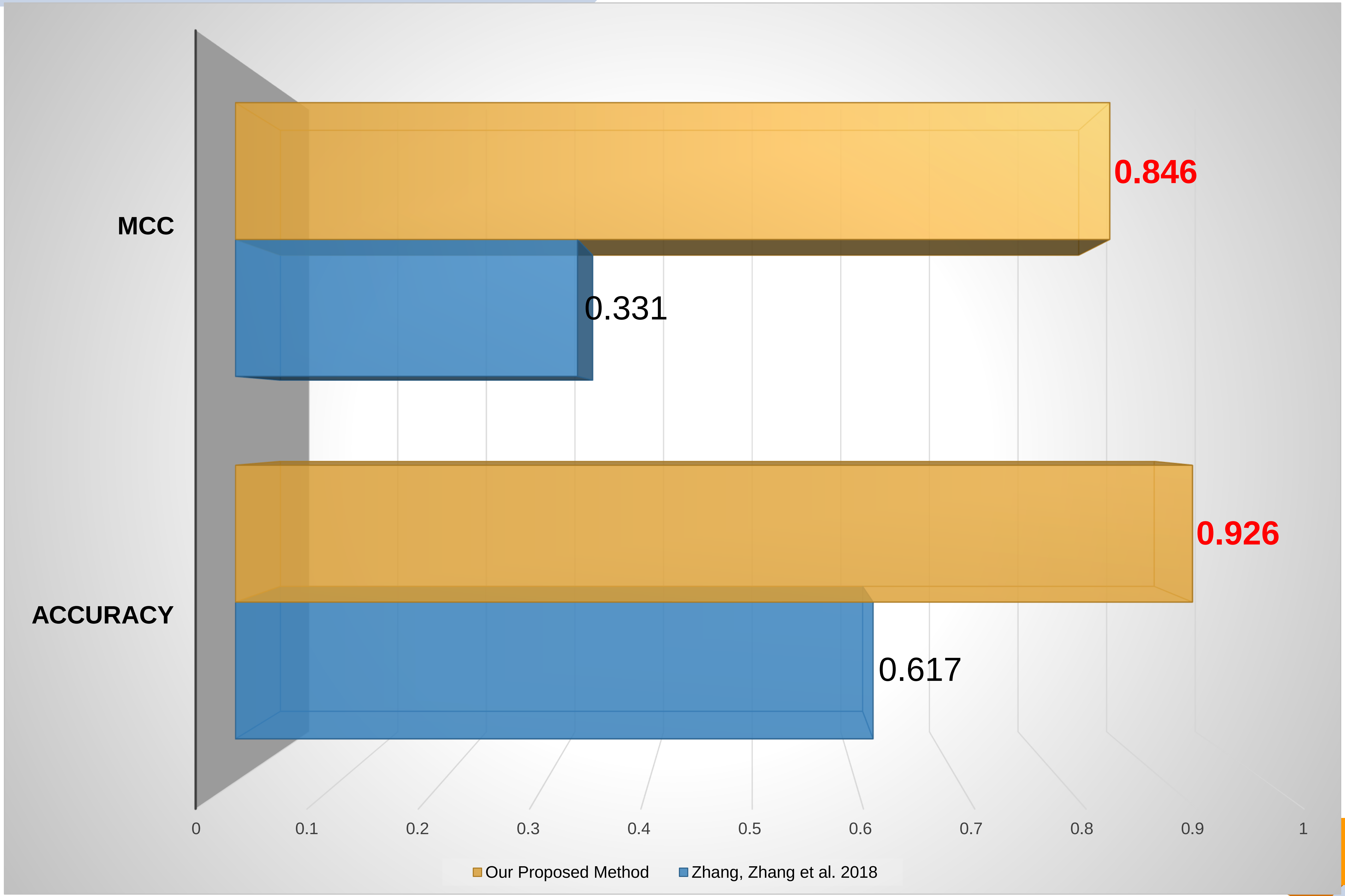
Comparison - Khaidem, Saha et al. 2016 - GE







Comparison – Zhang, Zhang et al. 2018



CONCLUSION

- **Our proposed method** provide **highly accurate** forecast **compare to the other** existing methods.
- Model using **long-term trading days' period** with CNN learning algorithm **achieves** the **highest performance** of sensitivity, specificity, accuracy, and MCC.
- Adding the indicator such as **volume** in candlestick chart **not** really help the algorithms **increase** finding the hidden pattern.

THANKS!