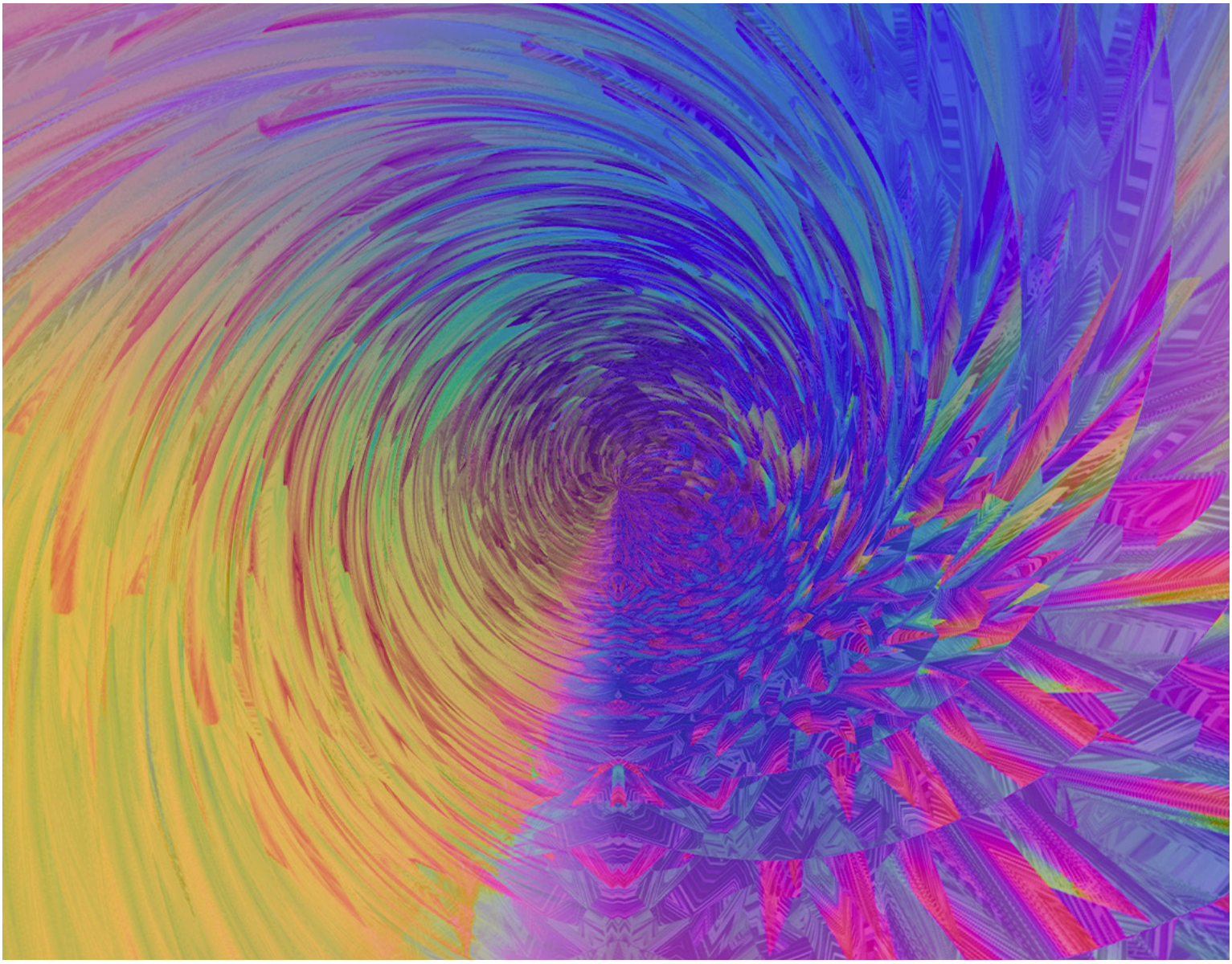
ISSN No: 2582-8312

Lattice

THE MACHINE LEARNING JOURNAL

VOLUME-2, **ISSUE**-4 (October - December 2021)



AN INTERNATIONAL PEER REVIEWED JOURNAL
IN DATA SCIENCE & MACHINE LEARNING

ADaSci | THE ASSOCIATION OF DATA SCIENTISTS

www.adasci.org

Index

About THE JOURNAL

Lattice is an international peer-reviewed and refereed journal on machine learning. The journal is hosted and managed by the Association of Data Scientists (ADaSci). Lattice intends to publish high-quality research articles of the researchers and professionals working in the field of data science and machine learning. All the articles published by Lattice have to pass an in-depth doubly blinded review process before publishing. The journal maintains a list of reviewers and editors all belonging to the prestigious institutions/organizations that take part in the functioning of the journal.

Scope OF LATTICE

The Association of Data Scientists was formed with the intent to develop, disseminate and implement knowledge, basic and applied research and technologies in analytics, decision-making, and management. Lattice, hosted under the flagship of ADaSci follows the same vision and aims to provide a platform for sharing and exchanging knowledge and research outcomes in the field of data science and machine learning. Lattice publishes scholarly articles that come under the aim and scope of the journal. The article submitted for consideration must consist of new concepts, theories, methodologies, and applications that are unpublished. It considers articles from the following key areas of data science and machine learning.

Ethics Concerns (PLAGIARISM, MISCONDUCT, ETC.)

The publication of an article in Lattice is considered as a building block in the development of a coherent and respected network of knowledge. The publication is a direct reflection of the quality of the contribution of an author and the organization that supports them. It is, therefore, necessary to adhere to certain standards of expected ethical behaviour. The important points that must be considered before submission are given below.

AUTHORSHIP: Authorship should be limited to those persons only who have made a significant contribution to the conception, design, execution, or interpretation of the reported study. Transparency about the contributions of authors is encouraged.

ORIGINALITY AND PLAGIARISM: The authors should ensure that they have written entirely original works, and if the authors have used the work and/or words of others, that this has been cited or quoted appropriately.

DATA ACCESS AND RETENTION: Authors may be required to provide the raw data in connection with a paper for editorial review, and should be prepared to provide public access to such data.

ACKNOWLEDGEMENT OF SOURCES: Proper acknowledgement of the work of others must always be given. Any funding received for the research must also be acknowledged.

DISCLOSURE AND CONFLICTS OF INTEREST: All submissions must include disclosure of all relationships with any member of the Lattice's editorial team that could be viewed as presenting a potential conflict of interest.

Copyright POLICY

The Lattice requires the transfer of copyrights from the author to the journal. On successful acceptance of every paper to the Lattice, the authors are required to submit a copyright transfer form. The copyright is transferred from the author to the publisher that is meant for the contents in the article. The algorithms and research work is always the intellectual property of the researcher or writer. Consider the case that in future, the author claims that ADaSci has published my work without my knowledge and consent or the author publishes the work with any other publisher and the other publisher claims that ADaSci has published its contents by violating the copyright rules.

The copyright aims to ensure that the researcher has published his work with the publisher to whom the researcher has transferred the copyrights. Now the publisher is the owner of the contents and the publisher of the intellectual property that actually belongs to the researcher. After getting the copyrights transferred from the author, the publisher becomes authorised to publish the contents on his publication mediums such as website, journal, video series etc. The copyright also ensures that the same content is not being published by the author with any other publication without the consent of the current publisher. It also ensures that no other person can publish the contents which are already published where the publisher has the copyrights for the same.

Disclaimer

The Association of Data Scientists (ADaSci) believes that the manuscripts submitted to Lattice by the corresponding authors are their original work as the authors have acknowledged the same while transferring the copyright to the journal. In future, if it is found that the content has been published with any other publication without the knowledge of ADaSci, the Lattice will discontinue the publication of that manuscript from the website.

The Editorial Board

The Editorial Board

List of Papers

List of Papers

Training Efficient CNNS: Tweaking the Nuts and Bolts of Neural Networks for Lighter, Faster and Robust Models

Sabeesh Ethiraj
UpGrad Education Pvt Limited,Mumbai
sabeesh90@yahoo.co.uk

Bharath Kumar Bolla
Salesforce, Hyderabad
bolla111@gmail.com

***Abstract*—**Deep Learning has revolutionized the fields of computer vision, natural language understanding, speech recognition, information retrieval and more. Many techniques have evolved over the past decade that made models lighter, faster, and robust with better generalization. However, many deep learning practitioners persist with pre-trained models and architectures trained mostly on standard datasets such as Imagenet, MS-COCO, IMDB-Wiki Dataset, and Kinetics-700 and are either hesitant or unaware of redesigning the architecture from scratch that will lead to better performance. This scenario leads to inefficient models that are not suitable on various devices such as mobile, edge, and fog. In addition, these conventional training methods are of concern as they consume a lot of computing power. In this paper, we revisit various SOTA techniques that deal with architecture efficiency (Global Average Pooling, depth-wise convolutions & squeeze and excitation, Blurpool), learning rate (Cyclical Learning Rate), data augmentation (Mixup, Cutout), label manipulation (label smoothing), weight space manipulation (stochastic weight averaging), and optimizer (sharpness aware minimization). We demonstrate how an efficient deep convolution network can be built in a phased manner by sequentially reducing the number of training parameters and using the techniques mentioned above. We achieved a SOTA accuracy of 99.2% on MNIST data with just 1500 parameters and an accuracy of 86.01% with just over 140K parameters on the CIFAR-10 dataset.

Keywords—Efficient Deep Learning, Mosaic ML, Depth Wise Separable convolutions, Stochastic Weight Averaging, Blurpool, Mixup, Cutouts, Label Smoothing, Sharpness Awareness Minimization, One Cycle LR, Global Average Pooling

I. Introduction

For the past decade, deep learning with neural networks has been the most popular way for training new machine learning models. The ImageNet competition in 2012 is widely credited with its rise to fame. A deep convolutional network AlexNet [1] fared 41% better than the next best entry that year. As a result of this ground-breaking study, a race to build deeper networks with an ever-increasing number of parameters and complexity ensued. Several model architectures, including VGGNet [2], Inception [3], ResNet [4], and others, have consistently beaten prior marks in ImageNet contests over the years while also expanding their footprint (model size, latency, etc.). Progressive improvements on benchmarks like image classification, text classification, and so on have been correlated with an increase in network complexity, the number of parameters, amount of training resources required to train the network, prediction latency, and so on since deep learning research has been focused on improving the state-of-the-art.

While these models can perform effectively on the tasks for which they were trained, they may not be suitable for immediate deployment in the real world. Building efficient deep learning models is imperative given the hardware and practical constraints. Efficient models can be broadly classified based on two categories; 1. Inference efficiency and 2. Training efficiency. Inference efficiency deals with how many parameters the model has, how large the model is, how much RAM is consumed during inference, how long the inference delay is, etc. On the other hand, training efficiency deals with how long it takes for a model to train, how many devices there are, if the model is memory efficient, etc.

However, we may not need to optimize for any given scenario for both types of efficiencies. In this paper, we have revisited techniques that pertain to increased inference efficiency. Our work will help the industrial and academic community to get introduced to the latest techniques and help them implement the same in their field of work

The objective of this paper is as follows:

1) To make models more inference efficient by improving architectural efficiency.

2) To make models more robust by data augmentation.

3) To improve model generalizability by tweaking optimization algorithms, manipulating labels, and altering weight space.

II. Related work

Much work has been done in the preceding years on making models more and more efficient as the impact of deep learning models has become increasingly evident lately.

A. Architectural Efficiency

The role of depth-wise convolutions in improving architectural efficiency was first established in the Xception network [5]. Depth-wise, separable convolutions were later

incorporated in the MobileNet [6] architecture to build lighter models. Depth-wise separable convolutions decrease the number of training parameters without compromising the dimensionality of the features/channels extracted compared to a conventional 3x3 kernel. While old school CNN architectures use dense layers at the end of convolutional blocks to cater for increased learning ability via an increase in parameters, this leads to flattening of the network, which compromises a neural network's ability to localize the features extracted by the preceding convolutional blocks as reflected in work done by [7]. Similarly, the work done by [8] showed that while the earlier layers capture only low-level features, the higher layers capture task-specific features. There is a need to preserve these features extracted, which were made possible by the concept of Global Average Pooling, wherein the information from the convolution blocks are condensed via averaging and are at the same time linearized before classification. Though feature extraction and preservation is an inherent task of convolutional layers, techniques such as max-pooling result in the loss of shift equivariance due to sub-sampling. To counter this, the concept of anti-aliasing filters was introduced in the pioneering work by [9] that could render a neural network more shift-invariant, subsequently making the model more robust. The deeper layers of a neural network are highly dimensional. However, only specific channels are hypothesized to contain related yet vast amounts of information for a given classification task at a given depth. Hence attention to these specific channels must be given if a deep network is expected to come out with a definitive output. The same was elaborated in work done by [10] wherein the channel outputs were passed through a sequence of dense layers with sigmoid activation to arrive at different scalar weights, which are later applied to the corresponding channels in a '*squeeze and excite*' manner.

B. *Data Augmentation*

Several data augmentation techniques have been innovated over the last decade, resulting in increasing a model's overall performance. State of the art results was obtained on CIFAR-10, CIFAR-100, and SVHN datasets using techniques such as cutouts [11], where random regions of an input image were masked out during training to make the models more accurate, robust, and generalizable. Similar work on augmentation done by [12] introduced the concept of mixup where input images and corresponding target variables were combined in varying proportions, resulting in completely new virtual training data used to improve model robustness.

C. *Optimization and Regularization*

In any neural network, the outcome that determines the accuracy of a model are fundamentally the weights that are learned over a considerable number of epochs. The work done by [13] showed that simply averaging the weights over a constant or a cyclical learning rate schedule resulted in better model performance. 75% of the training is done using the conventional training procedure, after which stochastic weight averaging is done as per the learning rate schedule. Another research done along the lines of optimization was the sharpness awareness minimization by [14]. Instead of just minimizing the training loss, the sharpness of the loss in neighboring loss space was also minimized, preventing the model from reaching localized minima, resulting in more generalizability. The revolutionary work [15] on label smoothing in a multi-class scenario where a fraction modified the target variable revealed that this technique prevented a model's overconfidence, subsequently increasing its robustness.

III. METHODOLOGY

State-of-the-art techniques such as Architectural efficiency enhancement, Weight Space Alterations, Label manipulation, Optimization, and Faster training have been used in the Mosaic ML library experiments.

A. *Dataset*

MNIST and CIFAR 10 datasets have been used to conduct the experiments mentioned. The datasets consist of 50000 training and 10000 test data points.

B. *Parameter Reduction - Depth Wise Separable Convolutions*

Depth-wise convolutions reduce the number of training parameters due to their mathematical efficiency of depth-wise channel separation and point-wise convolutions. A sample representation of depth-wise separable convolutions with a 3x3 filter on a 3-channel image is shown in Fig 1, showing a reduction in the number of parameters.

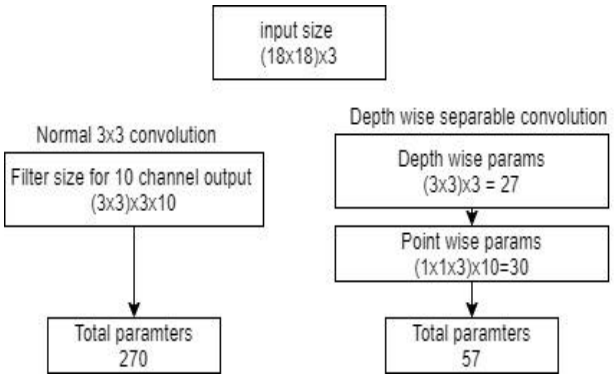


Fig 1. Depth Wise Separable Convolutions

C. *Parameter Reduction - Global Average pooling*

Global average Pooling (GAP) is performed by taking the average of all the neurons/pixels in a channel for all the channels to linearize the output of a convolution.

D. *Channel Attention - Squeeze and Excite*

Squeeze and Excitation (SE) blocks consist of two separate mechanisms which add attention to specific channels from the output of the previous convolutional layer. The initial squeeze mechanism is done by performing a Global Average Pooling to the output channels followed by an excitation mechanism performed by adding MLP layers with a sigmoid activation to generate a linear scalar weight. These scalar weights are applied to the feature maps or channels to generate the output of the SE block. In our experiments, we have added the SE block at the end of every convolutional layer using appropriate '*latent channel*' and '*min channel*' hyperparameters of the Mosaic ML library.

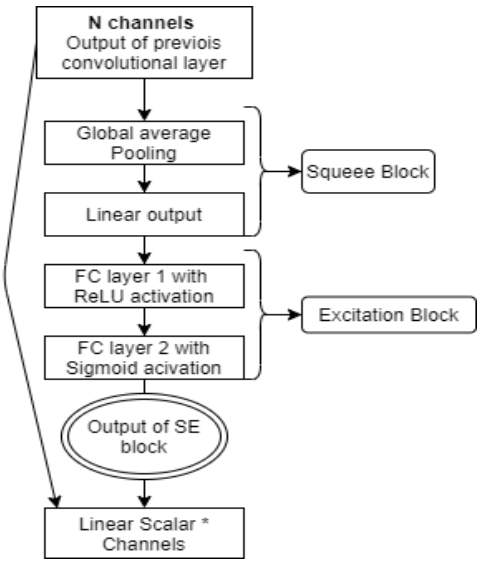


Fig 2. Squeeze and Excitation Blocks

E. Weight Space Alterations – Stochastic Weight Averaging

Stochastic Weight Averaging (SWA) is a regularization method that works on the principle of cyclical learning rate. There are two models. The first model stores the average weight of the models at the end of each learning rate cycle schedule of the second model, while the second model runs the conventional training algorithm. The final predictions are based on the stochastic weights average stored in the first model. In our experiments, we have implemented SWA after 75% of the training time, after which the weights are averaged after every training epoch

F. Anti-Aliasing techniques – Blurpool

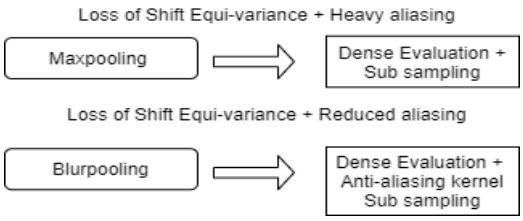


Fig 3. Blurpool

Convolutions and Maxpooling are susceptible to translational or shift variance of the input image as they, in a manner, cause downsampling of the input data resulting in aliasing of the input signals. Blurpool overcomes these deficiencies by introducing anti-aliasing at the time of downsampling. While conventional max-pooling is considered a combination of densely evaluated pooling and subsampling, blur-pooling is done by introducing a blur filter after the densely evaluated pooling. This prevents loss of information resulting in anti-aliasing and resilience to shift-variance. The proposed methodology is described in Figure 3. We have implemented this technique in our experiments both at the convolutional and max-pooling layers

G. Cutout

The cutout is a regularization or augmentation technique performed by clipping pixels from the input image. This results in better learning as this helps regularize the model. We have implemented this technique by clipping random masks of the input image of dimension 10x10 pixel.

H. Mixup

Mixup is a regularization technique that works on the principles of combining different input samples along with their target labels to create a new set of virtual training examples. The amount of mixup between the images and the labels is controlled by a hyperparameter δ which ranges between 0 and 1. The mathematical intuition of this technique taken from the original paper is shown below in Equation 1 where $\hat{x}$ and $\hat{y}$ are new virtual distributions created from existing training images and their corresponding labels.

$$\hat{x} = \delta x_i + (1-\delta)x_j$$
$$\hat{y} = \delta y_i + (1-\delta)y_j$$

Equation 1. Mixup Virtual distribution

I. Sharpness Awareness Minimization

Sharpness Awareness minimization is an optimization algorithm that minimizes both the training loss and the loss sharpness. Instead of minimizing just the loss, which may result in the gradient descent algorithm approaching the local minima, the SAM aims to approach the global minima whose neighborhoods have uniformly low training values in a given loss space. This results in better model generalization. Visualization of the loss landscapes as depicted in the original paper is shown in Figure 4.

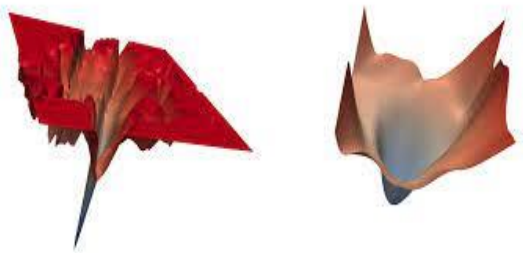


Fig 4. Sharpness Aware minimization with uniformly low neighborhood training loss (Right)

J. Label smoothing

Label smoothing is a regularization technique that changes the predicted target variable by a small quantity α. This prevents the model from overconfidence of a particular predicted target variable. A hyperparameter α controls the smoothening. In our experiments, we run with α = 0.1 to study the least possible label smoothing effect

K. One Cycle Learning Rate

In this technique, the learning rate is gradually increased from a minimum learning rate to a maximum learning rate during the first half of the cycle and then brought down to a value lower than the initial lower learning rate during the second half of the cycle. The higher learning rate towards the middle of the training helps prevent a sharp decrease in loss and subsequent overfitting. In contrast, the reduction in learning rate towards the end of the second half of the cycle helps achieve the local minima of the loss space much more efficiently. The change in learning rate takes place after every batch.

Table 1. MNIST model architectures

GAP Models	25 K without depth-wise convolutions	7K without depth-wise convolutions	7k with depth-wise convolutions	5K without depth-wise convolutions	5K with depth-wise convolutions	1.5K with depth-wise convolutions
Params	25,144	7,767	7,611	5,712	5,616	1,560
Layers	Conv 1 (1,8) /3x3 Conv 2 (8,16) /3x3 Maxpool1 Conv 3 (16,32) /3x3 Conv 4 (32,64) /3x3 Maxpool2 GAP Conv 5 (64,10) /1x1 (*Classifier)*	Conv 1 (1,8)/3x3 Conv 2 (8,12)/3x3 TB Conv (12,8)/1x1 Maxpool1 Conv 3 (8,12)/3x3 Conv 4 (12,16)/3x3 TB Conv (16,12)/1x1 Maxpool2 Conv 5 (12,15)/3x3 Conv 6 (15,15)/3x3 GAP Conv 7 (15,10) /1x1 (*Classifier)*	Conv 1 (1,10)/3x3 **Conv 2 DW (10,12)/3x3** TB Conv (12,10)/1x1 Maxpool1 Conv 3 (10,13)/3x3 Conv 4 (13,16)/3x3 TB Conv (16,12)/1x1 Maxpool2 Conv 5 (12,15)/3x3 Conv 6 (15,15)/3x3 GAP Conv 7 (15,10) /1x1 (*Classifier)*	Conv 1 (1,8)/3x3 Conv 2 (8,12)/3x3 TB Conv (12,8)/1x1 Maxpool1 Conv 3 (8,12)/3x3 Conv 4 (12,16)/3x3 TB Conv (16,12)/1x1 Maxpool2 Conv 5 (12,15)/3x3 GAP Conv 6 (15,10) /1x1 (*Classifier)*	Conv 1 (1,8)/3x3 **Conv 2 DW (8,12)/3x3** TB Conv (12,8)/1x1 Maxpool1 Conv 3 (8,13)/3x3 Conv 4 (13,18)/3x3 TB Conv (18,12)/1x1 Maxpool2 Conv 5 (12,16)/3x3 GAP Conv 6 (16,10) /1x1 (*Classifier)*	Conv 1 (1,8)/3x3 **Conv 2 DW (8,12)/3x3** TB Conv (12,8)/1x1 Maxpool1 **Conv 3 DW** (8,12)/3x3 **Conv 4 DW** (12,16)/3x3 TB Conv (16,12)/1x1 Maxpool2 **Conv 5 DW** (12,15)/3x3 GAP Conv 6 (15,10) /1x1 (*Classifier)*
DW	No	No	Yes	No	Yes	Yes
GAP	Yes	Yes	Yes	Yes	Yes	Yes

(*TB – Transition block / DW – Depth wise separable convolutions/ GAP – Global Average Pooling layer)

L. MNIST Model Architectures

Four different model architectures were built, sequentially reducing the number of parameters using depth-wise separable convolutions and Global Average Pooling layer; Model with 25K params, 7K params, 5K params, and 1.5K params. The summary of the architectures is shown in Table 1.

M. CIFAR 10 Model Architectures

Table 2. CIFAR-10 architectures

GAP Models	143 K Without DW	143 K with DW
Params	143,208	143,396
Layers	Conv 1 (3,16) /3x3 Conv 2 (16,16) /3x3 Maxpool1 Conv 3 (16,32) /3x3 Conv 4 (32,32) /3x3 Maxpool2 Conv 5 (32,64) /3x3 Conv 6 (64,64) /3x3 Maxpool3 Conv 7 (64,120) /3x3 GAP Conv 8 (120,10) /1x1 (*Classifier)*	Conv 1 (3,16) /3x3 Conv 2 (16,16) /3x3 Maxpool1 Conv 3 DW (16,32) /3x3 Conv 4 (32,32) /3x3 Maxpool2 Conv 5 DW (32,64) /3x3 Conv 6 (64,64) /3x3 Maxpool3 Conv 7 DW (64,128) /3x3 Conv 8 DW (128,192) /3x3 Conv 9 DW (192,260) /3x3 GAP Conv 2 (260,10) /1x1 (*Classifier)*
DW	No	No
GAP	Yes	Yes

In the case of CIFAR-10 experiments, we have built two models with Global Average Pooling Layers. The summary of the architectures is shown in Table 2.

N. Experimental Methodologies

Building these architectures is to sequentially reduce the number of parameters using GAP and depth-wise separable convolutions and to arrive at a model with the least number of parameters with a balanced accuracy – parameters trade-off. These models are later used to perform experimental studies using the various model enhancement techniques mentioned above by studying the effect of these using metrics such as Accuracy, inference time and, model size. The inference time of the models is calculated on the test dataset of 10,000 data points using the CPU as the inference engine.

O. Loss Function

The loss function used here is the cross-entropy loss as this is a multi-class classification scenario

$$CE = -\sum_{i}^{C} t_i log(s_i)$$

Equation 2. Cross-Entropy Loss

P. Evaluation Metrics

The evaluation metrics used in this study are Validation Accuracy, Inference time, and Model size.

IV. RESULTS

A. The efficiency of Depth Wise Convolutions on Accuracy

From Tables 1,2, and 3, it is evident that depth-wise (DW) convolutions in combination with GAP help reduce the number of training parameters, but do they actually help increase model performance? It has been one of the prime objectives of this research. To test this hypothesis, i.e., the efficiency of the network in terms of accuracy, the number of parameters is increased in our DW CNNS to match the CNNs without DW convolutions. It is observed that the addition of DW CNNs does not compromise model performance in terms of accuracy despite the reduction in the number of parameters; instead, they perform better or equally well as that of a network with 3x3 convolution in 25K, 7K, and 5K models. This is evident when we achieved a **SOTA accuracy of 98.35% in the architecture with 1.5K parameters** which are as good as a model with 7K or 25K parameters. Similar performances were also seen on the CIFAR-10 dataset, where the model performed equally well as that of a 3x3 convolutional network. Hence, it can be inferred that **depth-wise separable convolutions can help achieve higher**

accuracy with fewer parameters. Furthermore, there is also a reduction in model size using DW convolutions due to a reduction in the number of parameters, as seen in Table 3

Table 3. MNIST /CIFAR10 - Accuracy, Latency, and Model Size

MNIST Dataset				
Model	DW	Acc	Latency	Size
25K (25,144K)	No	99.35	3.36 s	107 KB
7K (7,767K)	No	99.34	2.67 s	43 KB
7K (7,611K)	Yes	99.46	3.35 s	43 KB
5K (5,712K)	No	99.33	2.76 s	34 KB
5K (5,616K)	Yes	99.27	2.74 s	34 KB
1.5K (1,560K)	**Yes**	**98.35**	**2.52 s**	**19 KB**
CIFAR 10 Dataset				
143K (143,208K)	No	81.57	8.51 s	578 KB
143K (143,396K)	Yes	79.9	8.82 s	590 KB

B. Effect of Depth Wise Convvolutions on Inference Time

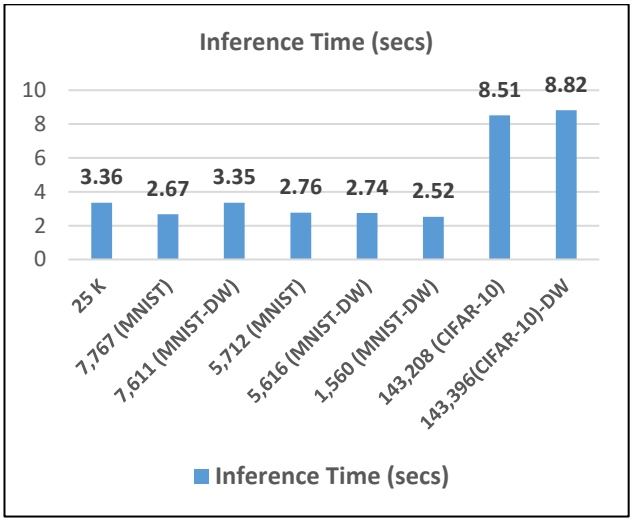


Fig 5. Inference time of Various models - 10000 Test Datapoints

Depth wise convolutions do not have a significant beneficial effect on the inference time of the model. In fact, models with **depth-wise convolutions perform slower than models with 3x3 kernels**. This is evident on both the MNIST and CIFAR-10 datasets, where we see increased latency (Table 3). It is hypothesized that this may be attributed to the two separate convolutional operations (depth-wise separable + point-wise convolution) that consume more mathematical computing time than a single 3x3 convolution. But to achieve faster and efficient models with depth-wise convolutions, there has to be a reduced parameter count and as seen in our experiments, the model with the least number of parameters **(1.5K) using depth-wise convolutions** has the **lowest latency of 2.52 seconds**. Thus, it can be inferred that depth-wise convolutions can help lower inference times by decreasing the number of parameters.

C. Model Efficiency Techniques

Models with the least number of parameters derived using depth-wise convolutions are subjected to model efficiency techniques mentioned above in the research methodology section. In general, these techniques improve model performance in terms of overall accuracy, with **Blurpool** being the most efficient and consistent of them. Table 4 below shows the improvement seen in the accuracy for various techniques that have been implemented in our work.

Table 4. Effect of various techniques on model accuracy

Methodologies using Depth-wise convolutions and GAP	Validation Accuracy	
	MNIST	CIFAR10
	1,560 Params	143,396 params
Baseline Model	98.35	79.9
One Cycle LR	98.92	79.42
Cutout (CO)	98.45	82.92
Blurpool (BP)	**99.21**	83.16
Squeeze and excite (SE)	99.18	81.61
Mixup (M)	97.85	**83.52**
Label Smoothing (LS)	98.83	81.06
Sharpness aware minimization (SAM)	98.77	80.64
Stochastic weight averaging (SWA)	99.02	82.1
BP + SE + SWA	**99.2**	-

Table 5. Combination of Techniques on Accuracy - CIFAR 10

Combination techniques using Depth-wise convolutions and GAP – CIFAR10	Validation Accuracy
BP + CO + M + SE	85.55
BP + CO + M	**86.08**
BP + CO + M + SAM	86.09
BP + CO + M + SWA	86.37
BP + CO + M + LS	86.37
BP + CO + M + LS + SWA + SAM	**86.76**

Performance on MNIST

Blurpool is the most efficient model enhancement technique in the MNIST dataset, where there was an increase in accuracy from 98.35% (baseline) to **99.21%** with just **1.5K** training parameters. Furthermore, the top three performing techniques (Blurpool, Squeeze and excite, Stochastic Weight Averaging) were further combined to study the performance improvement; however, the accuracy remained static at 99.2% (Table 4). Hence no further combination techniques were experimented upon.

Performance on CIFAR-10

Similar performance improvements were noted on the CIFAR-10 dataset with techniques such as Mixup, Blurpool, and Cutouts. **Mixup fared better** with a **3.62 % increase** in accuracy from 79.9% to 83.52%.

Inspired by this leap in accuracy with just a single technique, we further combined the top-performing techniques in varying combinations to push the model to attain higher accuracies. It was seen that (Table 5), combining BP + CO + M + LS + SWA + SAM, resulted in the model reaching a **SOTA accuracy of 86.76%** (**6.86% increase**) with just over 140K parameters.

D. Training Curves

The training curves for various depth-wise convolution models are shown below.

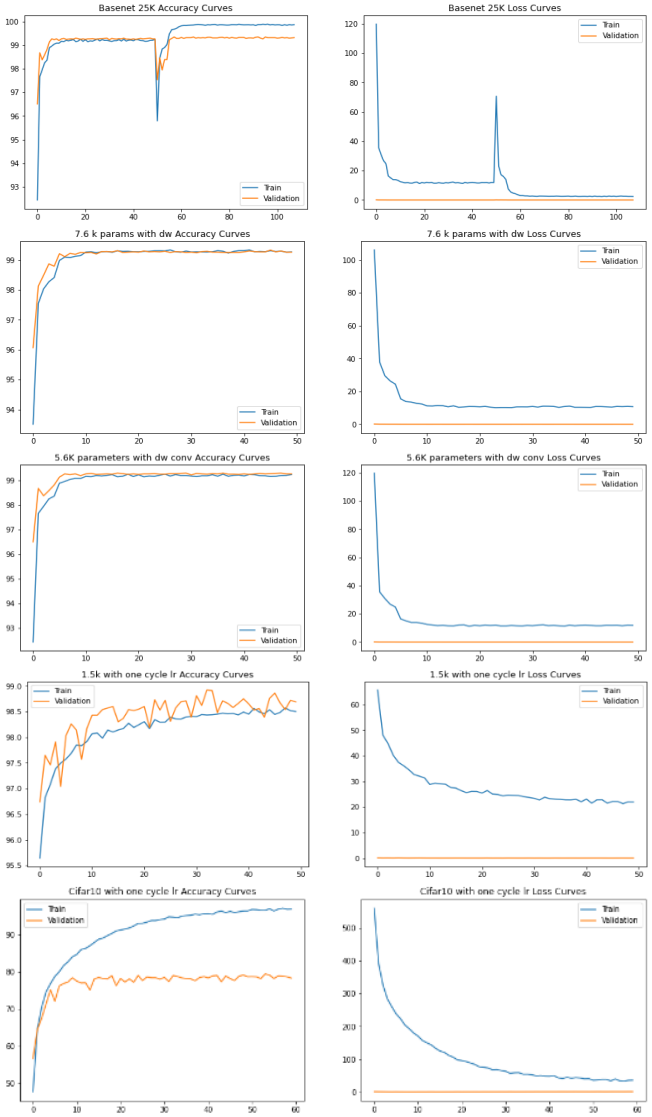


Fig 6. 25K, 7K,5K,1.5K (MNIST),143K model (CIFAR-10) (Top to Bottom) - Training curves

V. Conclusion

Our work sheds light on the efficiency of Depth wise separable convolutions and GAP. These techniques result in a significant reduction in the number of parameters. Though they cannot match the feature extraction capabilities of a conventional 3x3 kernel, fine-tuning the architecture with these techniques will result in equivocal if not better performances of these models, making them ideal for deployment on Edge and mobile devices. As seen in our study, we achieved a SOTA accuracy of 98.35% on the MNIST dataset using just over 1.5K parameters and a model size of 19 KB.

Furthermore, it was observed that combining various model enhancement techniques resulted in better accuracies, as in the case of MNIST, where we achieved an accuracy of 98.35 % using Blurpool. On CIFAR10, the performance gains are even more substantial, with Cutout, Mixup, Blurpool, Label Smoothing, Stochastic Weight Averaging, and Squeeze & Excitation block applied in isolation. Combining all the six techniques, resulted in a phenomenal 6.86% increase from 77.9% to 86.76%. Though Depth wise convolutions do not directly affect the inference time of a model, their ability to reduce the number of parameters helps decrease the overall mathematical computational time required at the time of inference, hence speeding up the model.

References

1. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems 25 (2012), 1097–1105
2. Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014).
3. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition. 1–9
4. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770–778.
5. F. Chollet. Xception: Deep learning with depthwise separable convolutions. arXiv preprint arXiv:1610.02357, 2016.
6. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang,T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017
7. Lin, M.; Chen, Q.; and Yan, S. 2013. Network in network. arXiv preprint arXiv:1312.4400.
8. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A.2016. Learning deep features for discriminative localization. In Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, 2921–2929. IEEE.
9. Richard Zhang. Making convolutional networks shift-invariant again. In ICML, 2019
10. J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. arXiv preprint arXiv:1709.01507, 2017.
11. T. DeVries and G. W. Taylor. Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552, 2017
12. H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017.
13. Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. arXiv preprint arXiv:1803.05407, 2018.
14. Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. Arxiv, 2010.01412, 2020
15. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In CVPR, 2016

A case study on Credit Risk Analysis using Taiwanese Banking Data

Harshit Deepak Bhavnani

Mukesh Patel School of Technology Management and Engineering, NMIMS University, Mumbai

harshit.bhavnani@gmail.com

Shreyansh Suman Bardia

Mukesh Patel School of Technology Management and Engineering, NMIMS University, Mumbai

shreyanshbardia6@gmail.com

***Abstract*—This system-description paper essentially works towards aiding the financial industry in the sub-domain of credit risk by evaluating numerous machine learning algorithms in addition to neural networks, thereby observing that ensemble-based classifiers outperformed neural networks and the best performance was demonstrated by XGBoost classifier - a decision-tree-based ensemble machine learning algorithm that uses a gradient boosting framework that predicted with an accuracy score of 82.0833% and a precision score of 80.3417%. This research also led to an extensive survey of the socio-economical condition of Taiwan in order to in order to understand the relationship between the features present in the dataset and the results obtained. With this research, we also verified that the claims made by several researchers stating that gradient boosting and random forest algorithm is well suited for credit risk while neural networks may not give impressive results also holds true in this case.**

Keywords— ***Banking; Credit risk; Deep Learning; Machine learning; Neural Networks; Gradient Boosting***

I. Introduction

The coronavirus pandemic is a humanitarian crisis that led to a surge in unemployment. Despite efforts by several nations in providing stimulus packages to citizens and reduction in consumer spending, credit card debts have significantly increased on a global scale. For instance, credit card debt in the United States of America surpassed $1 Trillion in 2017[1]. Thus, the need for a robust classification system that ensures minimum defaults is of utmost importance.

According to statistics compiled by the Financial Supervisory Commission, in 2015, 36 banks issued credit cards in Taiwan and the number of cards in circulation was 38 million.[2] With such exponential growth of the credit card market, credit card defaults have a significant impact on the gross domestic product of Taiwan. For instance, during the 2005 credit card crisis of Taiwan, the losses of credit card debts were, based on the data from FSC, between NT$280 billion and NT$370 billion, or between 2.4% and 3.4% of Taiwan's GDP, larger than the subprime mortgage losses in the US[3]. High credit card indebtedness accounted for 9% of Taiwan's GDP in 2005.[4]

Major banks including Citibank, HSBC, Hua Nan Bank, and Land Bank of Taiwan have cut interest rates on credit card balances and are offering deferrals on credit card bill payments as the Taiwanese government is working towards promoting credit card usage. The government also launched a 'triple stimulus voucher' program offering cashback to customers on credit card purchases. Individuals from 38 banks can register for this program. Taiwan has been a cash dominated society. However, there has been a gradual shift towards digital payments supported by a high banked population, growing financial awareness and government push. This is becoming more prevalent now as people are avoiding cash payments for the fear of transmitting the COVID-19 virus. Card payments in Taiwan are expected to register an annual growth of 5.6% in 2020, as the COVID-19 pandemic has accelerated the push towards electronic payments. Card payments in Taiwan are expected to grow at a compound annual growth rate (CAGR) of 7.6% to reach TWD6.0 trillion (US$200.2bn) in 2024[5].

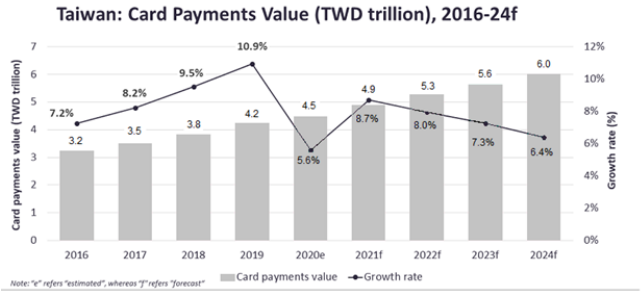


Figure 1: Projected growth in credit card usage

To be able to sustain in the current scenario, it has become imperative for banks to shift their focus from crisis management to risk prediction[6]. Risk prediction is conducted by leveraging information of the customer or organisation in order to predict their ability to pay back their debt in future. The practice of risk prediction in the banking industry has led to the inculcation of various machine learning and deep learning classification models. Some of these models are audited in this research in an attempt to find an efficient model for credit risk analysis that could be used by the banking industry.

In [11], the authors have stated that XGBoost and Random Forest have been found to be the most efficient classifiers in many cases while neural networks may not perform well at times. This was verified with our research.

The research paper is drafted with the aim of informing the readers with key definitions of all the algorithms used for the purpose of prediction. It also gives them a fair idea about the

previous work in the same field that have been used to derive significant insights for this research. Furthermore, they receive necessary information about the dataset that has been used. The data has been visualised to understand the impact of the predictors on the target variable effectively. The subsequent section explains the entire methodology of the research and the performance metrics used to evaluate the algorithms. The results of the evaluation are then tabulated and we finally conclude the best model.

II. Literature Review

This section is subdivided into the explanation of all the learning algorithms that have been used in addition to the inference derived by us from various research papers that have aided our work.

A. Key Definations

The classification algorithms that were used for the purpose of this research have been explained in this subsection.

1) Logistic Regression

Logistic regression is a linear algorithm that is used for performing binary classification. These models calculate the probability of the given input belonging to a particular class using the logistic function (also known as the sigmoid function) for classifying the data by converting mapping real values to a value between 0 and 1.

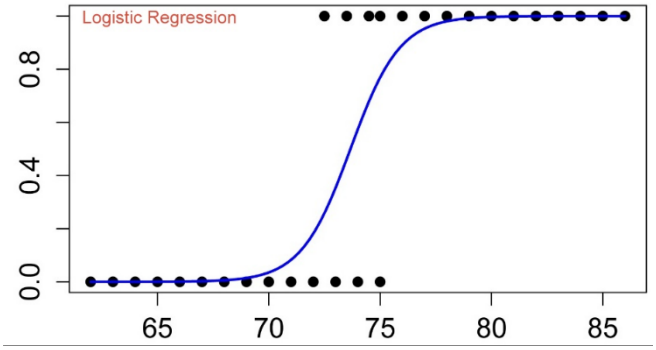


Figure 2: Sigmoid function of Logistic Regression

The formula of sigmoid function is:

$$z = \frac{1}{1+e^{-x}}$$

Logistic regression models are easy to implement but they have several implicit assumptions which may not be easily validated [12].

2) Random Forest Classifier

Random Forest is a supervised machine learning ensemble technique that is used to perform both binary as well as multiclass classification.

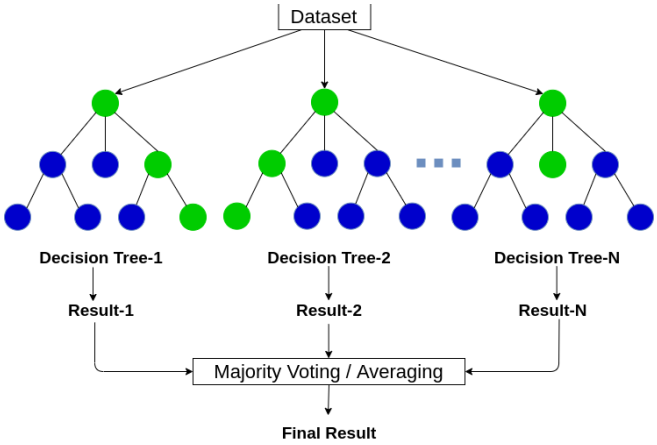


Figure 3: Pictorial Representation of Random Forest Algorithm

Random forests consist of several simple decision trees each of which gives an output for classifying the data to a particular class. The outputs are then aggregated to give the final result [13].

Random forests are one of the most popular machine learning algorithms as they can run efficiently on large datasets and are less prone to overfitting. However, they are slow to train and do not perform well on sparse and high dimensional datasets.

3) K - Nearest Neighbor Classifier

It is one of the simplest non-parametric algorithms to implement and it can be used for both regression and classification purposes. This technique calculates the distance between the test data and the nearest training data for determining the class of the test data.

A number of methods have been proposed to calculate this distance. Some of them are:

Manhattan Distance: $D = \sum_{i=1}^{n} |x_i - y_i|$

Euclidean Distance: $d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$

Minkowski Distance: $D = (\sum_{i=1}^{n} |x_i - y_i|^p)^{1/p}$

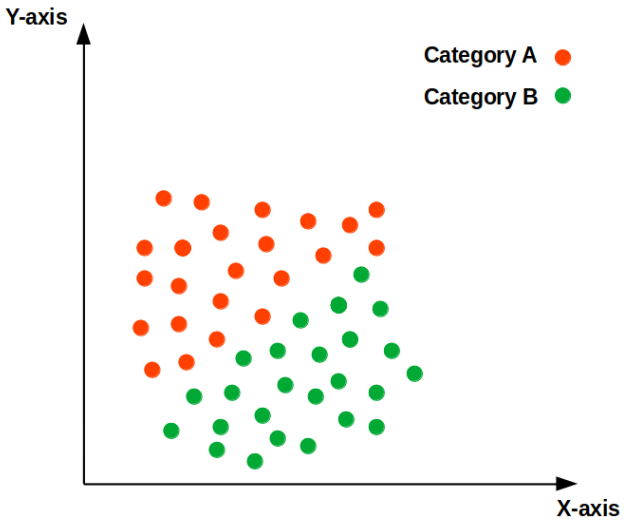


Figure 4: Pictorial Representation of K-NN algorithm

Although this technique is easy to implement, it requires a large amount of data which makes it computationally expensive [14],[15].

4) Support Vector Machine

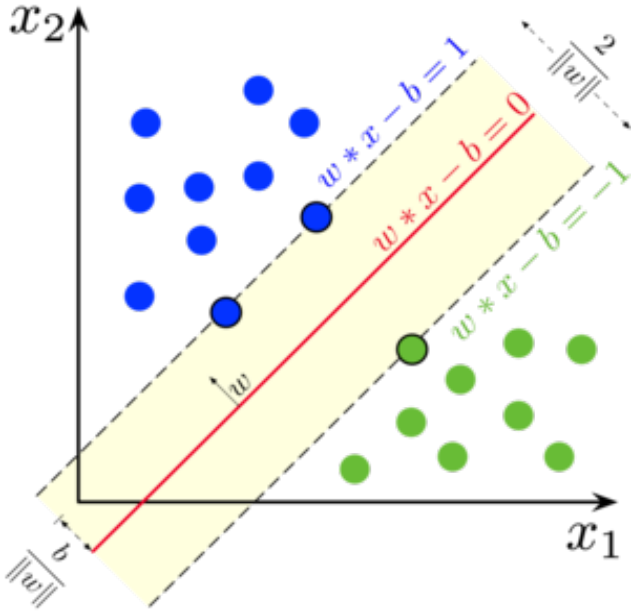


Figure 5: Pictorial Representation of SVM algorithm

Support Vector Machines (SVMs) perform classification as well as regression by making a decision boundary, known as a hyperplane, to classify the input data to a particular class.

SVM is an extremely popular machine learning algorithm as it has been known to provide results comparable to that of complex artificial neural networks and are less likely to overfit the data. There are several kernel functions developed that are used in SVMs for classifying both linear and non-linear data and they have been used for various tasks like handwriting recognition and facial analysis [16],[17].

SVMs are known to perform poorly when the classes are not well separated. Selecting the optimal hyperparameters of an SVM model can be difficult and selecting the right kernel function is also a tricky task.

5) Decision Tree Classifier

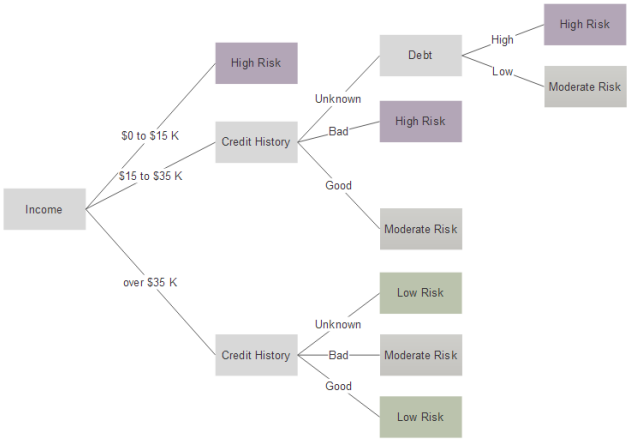


Figure 6: Pictorial representation of Decision Tree algorithm

Decision Tree algorithms have a structure consisting of internal nodes which contain the test data, branches that denote the outcome of the test and leaf nodes that contain the class label for the test data. These algorithms can be used for classification and regression. It is a popular technique for performing data mining [18].

Decision tree models are non-parametric models and the data on which they are applied require little to no pre-processing. However, they are more likely to overfit on the data, therefore, regularization of these models is necessary.

6) Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a dimensionality reduction technique, i.e.it is a technique that reduces the number of variables in a dataset while retaining as much information as possible.

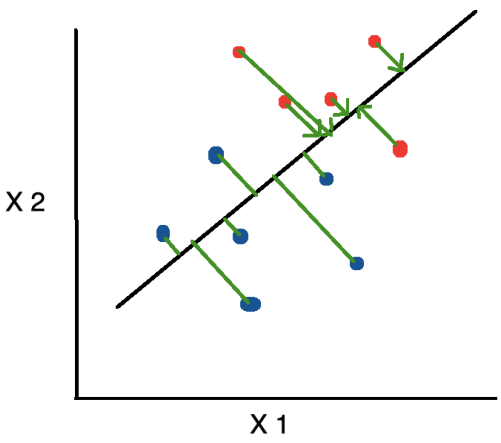


Figure 7: Dimension reduction using LDA algorithm

An example of this technique would be to flatten a two-dimensional dataset into a one-dimensional dataset and use 'within-class' and 'between-class' scatter matrices to compute eigenvectors from corresponding eigenvalues so that new features with fewer dimensions could be obtained. The model uses Bayes Theorem to estimate the probabilities [19].

7) Gaussian Naïve Bayes Classifier

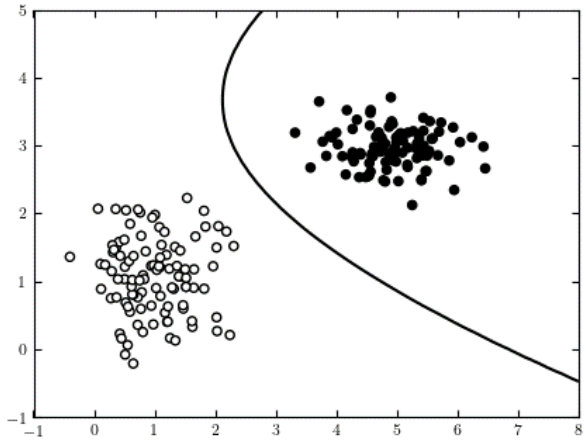


Figure 8: Pictorial representation of Gaussian Naïve Bayes Classifier

Naïve Bayes classifiers are a set of supervised machine learning algorithms that are developed on the Bayes theorem. These classifiers treat each feature as an independent variable.

Gaussian Naïve Bayes classifier is a variant of the naïve bayes classifier and is developed using Bayes theorem.

$$P = \frac{P(B|A).P(A)}{P(B)}$$

. It assumes a normal distribution of the data and is used on continuous data [20].

The classifier is easy to implement and requires a small amount of training data. However, it assumes that the predictor variables are mutually independent which is rarely the case in real-life situations.

8) Gradient Boosting Classifier

Gradient boosting is an ensemble technique consisting of gradient descent and boosting.

In this technique, a new weak learner is sequentially introduced for compensating for the errors of the existing weak learners. These errors are identified by the gradients [21].

Gradient boosted machines can be used for both classification and regression and their predictive accuracy is usually very high. They also have several hyperparameters for optimization making the model very flexible and can directly work on data having both numerical as well categorical features which reduces the amount of pre-processing to be performed on the data.

Some of the disadvantages of Gradient Boosted Machines are that they can be hard to tune and are prone to overfitting. They can become computationally expensive and the results obtained are less interpretable.

9) Extreme Gradient Boosting (XGBoost)

XGBoost is a variant of Gradient Boosting. It is a system that can be scaled as required for learning tree ensembles. It can also improve the hardware capabilities by optimizing cache and parallelizing the processes. It can also be regularized by methods like pruning and having sparsity pattern awareness. It is a flexible algorithm as its objective function as well as the evaluation metric can be customized.

This algorithm usually gives a very high accuracy and it also has a shorter training time as compared to other boosting methods. However, it does not perform well on regression tasks due to its poor ability to extrapolate values.

10) Artificial Neural Network

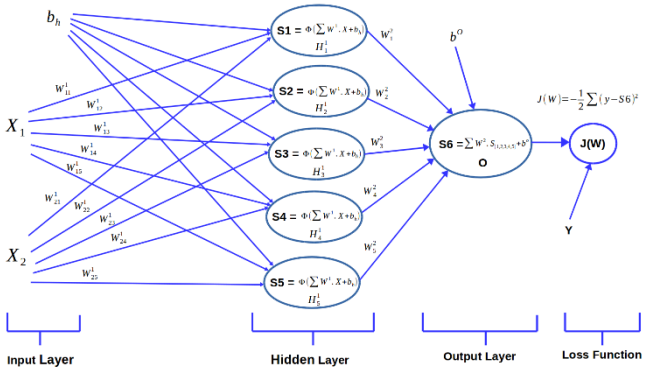


Figure 9: Structure of the ANN Classifier

An artificial neural network (ANN) is a deep learning technique that can be used for performing both pattern recognition and regression.

It is a mathematical model which consists of interconnected nodes called neurons. The number of neurons in a model depends on the complexity of the task for which the model is being used. For extremely sophisticated tasks such as facial recognition, the number of neurons can be in millions which gives them the capability to infer complicated patterns from the given data.

ANNs can do parallel processing which allows them to perform more than one job at a time. As a result, they can perform a large number of computations in a short period of time and give prediction results of very high accuracy [22],[23].

However, ANNs are computationally very expensive which necessitates having high-quality hardware resources to reduce the time taken to train the model. ANNs are black-box models. Although they may give the desired results for a given input data, the process by which the model arrived at those results is opaque and not clearly interpretable.

B. Related Works

Recently, researchers have tried to implement classical machine learning algorithms as well as deep learning models for performing credit risk analysis on different datasets such as the German and the Australian credit dataset [7,8]. Traditional machine learning models like logistic regression and XGBoost have been commonly used to develop credit risk prediction models. Due to the presence of severe data imbalance in such datasets, the researchers have compared the results on a few other metrics in addition to accuracy such as recall score, sensitivity, Area Under Curve and Root Mean Squared error [7,8].

New techniques have also been developed to detect credit card frauds such as the RUSMRN algorithm [9], which is developed on the AdaBoost.M1 technique and the data sampling technique. The RUSMRN algorithm also has the capability to handle data imbalance, thereby providing more reliable results.

Efforts have also been made to reduce the dimensionality of data and also develop credit rating systems based on machine learning models for assigning the consumers a credit score after calculating the probability of whether the consumer will default or not [10].

The financial industry commonly uses Machine Learning algorithms to derive the credit scores of customers. Research claims that generally, ensemble classifiers perform with maximum accuracy wherein XGBoost and Random forest are the most efficient and popularly used classifiers. Additionally, neural networks do not necessarily make better predictions [11]. This was verified with our research.

III. Dataset

The dataset used in this study is taken from the UCI Machine Learning Repository.[1] The study comprises payment data of a Taiwanese bank which was collected in the year 2005 and targets customers that hold credit cards of the bank. The dataset consists of 30,000 observations, out of which 6636 (22.12%) observations are observations of defaulters. The target variable (default.payment.next.month) is a binary variable which tells whether the customer has defaulted or not (Yes=1 and No=0).

TABLE I. Description of the Data

Sr. No.	*Attribute*	*Description*
1	ID	Numerical unique id for each customer
2	LIMIT_BAL	The credit limit which is given to each customer in NT dollars
3	SEX	Gender of the customers (1=Male, 2=Female)
4	EDUCATION	Education level of customers (1=Graduate School, 2=University, 3=High School and 4=Others)
5	MARRIAGE	Marital Status of customers (1=Married, 2=Single,3=Others)
6	AGE	Age of the customers in years
7	PAY_0 - PAY_6	History of repayments since 6 months
8	BILL_AMT1 -BILL_AMT6	Represents bill statement amount in NT dollars
9	PAY_AMT1 - PAY_AMT6	Represents the amount of money previously paid in NT Dollars
10	default.payment.next.month	Indicates whether a customer has defaulted on their payments for the next month or not

For the purpose of performance evaluation, 80% of the data (24,000 observations) is randomly selected to train all classifiers and the rest 20% of data (6000 observations) are tested.

IV. Methodology[2]

For the purpose of this research, our initial step is to pre-process the data. Once the data is ready for use, we build 10 models corresponding to the 10 statistical learning, machine learning and deep learning algorithms that have been chosen based on the difference in their nature from each other. The hyper-parameters of the most efficient model are tuned to develop the best model.

A. Data Pre-Processing

The column 'ID' was removed as it did not impact the target variable in any manner and its correlation with the same was insignificant. The column 'MARRIAGE' had categorical values that did not correspond to the description of the dataset. Thus, all extra values were combined to minimize redundancy.

After cleaning, the dataset was standardized in order to aid the efficacy of the results. The mathematical equation of the minmax scaler is: $x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$

where x is the single value from a column in our dataset and x-min and x-max are the minimum and maximum values in the same column, respectively.

B. Model Learning and Evaluation

For the purpose of learning, 9 classification algorithms were used, namely, Logistic Regression, Random Forest, K-nearest neighbour, Gradient Boosting, Extreme Gradient Boosting, Support Vector Machine, Decision Tree, Linear Discriminant Analysis, Gaussian Naive Bayes and artificial neural network.

[1] https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

[2] https://github.com/harshitbhavnani/Credit-Risk-Analysis-Project/blob/main/Credit_Risk_Analysis_Project.ipynb

The predictions made by the aforementioned algorithms were evaluated using 4 performance evaluation metrics - Accuracy score, Recall score, F1 score and Precision score.

C. Hyper-parameter tuning

Hyper-parameter tuning was carried out on the best performing classifier in order to get the most efficient model for classification.

D. Application

In an attempt to automate the process of predicting the tendency of a customer to default, the probability of defaulting predicted using the best model is multiplied by 100 in order to create a credit score for customers. A threshold could be set by the respective banks depending on their ability to take risks.

V. Experimental Study

This section entails information about performance metrics used in addition to accuracy score for addressing the class imbalance issue. It also consists of the results that we've achieved after the use of numerous classification algorithms and the performance of using an artificial neural network model. Lastly, it explains the impact of hyper-parameter tuning on the best classification algorithm among the ones that we have used.

A. Exploratory Data Analysis

The data was visualised in the research for understanding the impact of our variables on the payment status of the customers.

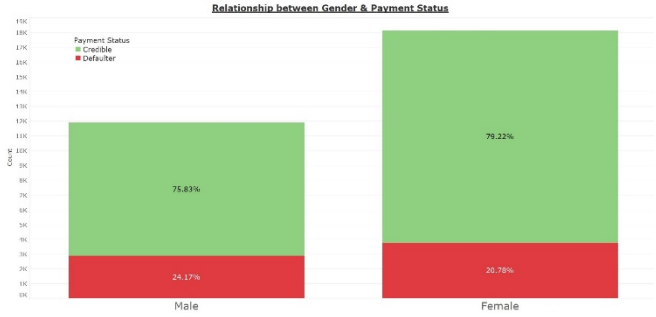


Figure 10: Relationship between Gender & Payment Status

The dataset consists of a larger number of women and the proportion of women who have defaulted are slightly higher compared to men. However, the inference does not seem conclusive.

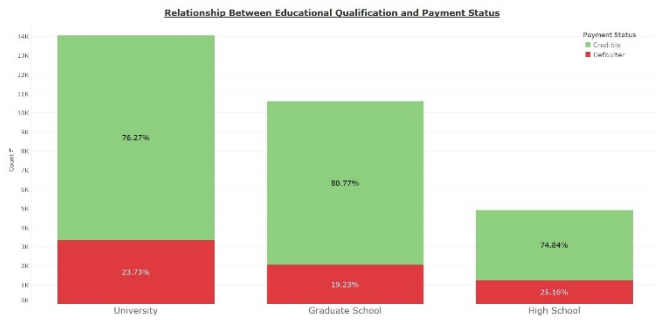


Figure 11: Relationship between Educational Qualification & Payment Status

The number of loans is seen to be maximum in case of customers who have been to university and minimum for those who have been to high school. We do not observe a major impact of the educational background on the payment status either. However, the credibility of the customer increases gradually with their educational background.

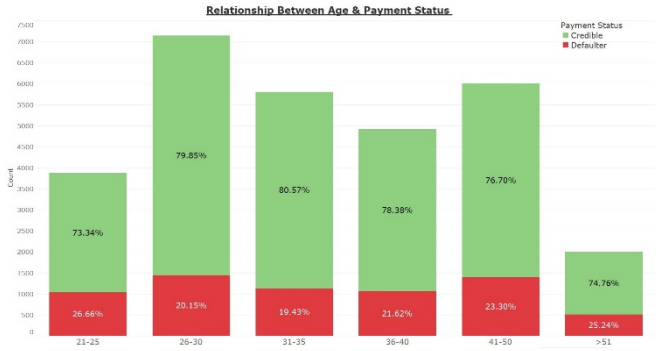


Figure 12: Relationship between Age & Payment Status

Maximum customers that have taken loans are between the ages of 26 to 30 and the minimum customers that take loans are seen to be beyond the age of 51. Individuals between the age group of 41 to 35 are seen to be most credible while those who are between the age group of 21 to 25 consist of the highest proportion of defaulters.

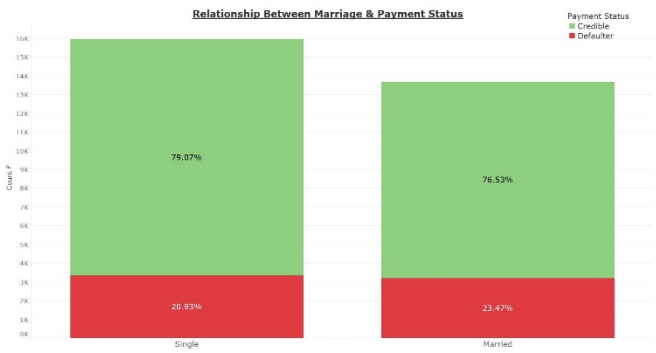


Figure 13: Relationship between Marriage & Payment Status

Single customers take more loans and are more credible than married customers.

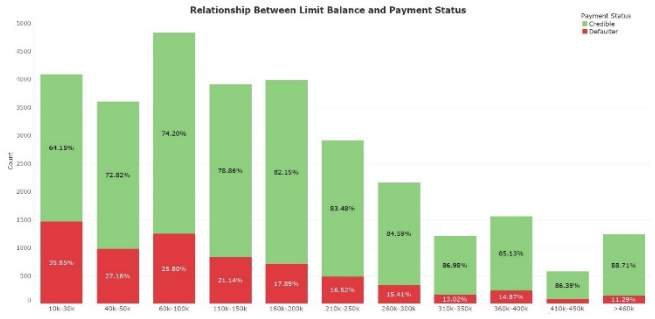


Figure 14: Relationship between Marriage & Limit Balance

We observe a strong relation between the limit balance that is allotted to the customers and their credibility. The customers that have been allotted a high limit balance have defaulted the least while the customers with low limit balance default the most.

B. *Performance Evaluation Metrics*

Since the dataset has a significant imbalance in the distribution of the prediction variable (i.e. the number of observations of non-defaulters are higher than the number of observations for defaulters), it may cause the problem of an accuracy paradox during the predictions. Thus, in order to efficiently evaluate the classifiers, different metrics are used that overcome this paradox and give better results. The metrics used apart from accuracy score are - Confusion matrix, Precision score, recall score and F1 score.

1) *Accuracy Score*

It is the ratio of the number of correct predictions to the number of total predictions made by the model.

$$Accuracy = (TP+TN)/(TP+TN+FP+FN)$$

Accuracy score is used for checking the number of accurate predictions out of total predictions made by the model. However, it is not considered as the ideal metric for gauging model performance for certain cases.

The data available for many domains including credit risk are severely imbalanced, meaning the amount of data for a particular class can be very high (majority class) as compared to the data available for the other class (minority class). This difference in data causes the model to become biased towards the majority class and the model misclassifies the minority class as the majority class.

Such misclassification can be a significant cause for concern for financial institutions because the minority class in the data available for credit risk analysis is the number of customers who would default, and the majority class would be the customers who would not default. In such cases, the cost of misclassifying the minority class as the majority class would be much higher as compared to the cost incurred if reverse misclassification occurs.

2) *Confusion Matrix*

A confusion matrix is a tabular structure that provides information about the accurate predictions as well the misclassifications made by the model. The actual values for each class are present along with the columns and the predicted values are present along the rows.

True Class \ Predicted Class	Positive Class	Negative Class
Positive Class	True Positive	False Negative
Negative Class	False Positive	True Negative

Figure 15: Components of Confusion Matrix

True Positive: The values which are correctly classified as the positive class are called true positives.

True Negative: The values which are correctly classified as the negative class are called true negatives.

False Positive: The values which belong to the negative class but are misclassified as the positive class are called false positives.

False Negative: The values which belong to the positive class but are misclassified as the negative class are called false negatives.

Note: The terms positive class and negative class are simply used to differentiate the two classes. There is no good or bad connotation associated with the names of the classes.

3) *Precision Score*

It is the ratio of the number of the values correctly classified as the positive class to the sum of the number of values correctly classified as the positive class and the number of values that belong to the negative class but are misclassified as the positive class.

The precision score is used for minimizing the number of false positives.

$$Precision\ Score = TP/(TP+FP)$$

4) Recall Score

It is the ratio of the number of values correctly classified as the positive class to the sum of the number of values that are correctly classified as the positive class and the number of values that belong to the positive class but are misclassified as the negative class. Recall score is used for minimizing the number of false negatives.

$$Recall = TP/(TP+FN)$$

5) F1 Score

It is a combination of precision and recall, and unlike its parent metrics, it is capable of giving us the complete picture. It is given by the formula:

F1-score = (2 x precision x recall)/(precision + recall)

C. Performance of Machine Learning Classifiers

After training all the classification models with 80% of the total data, these models are used to test the 20% of the total dataset that is remaining using the aforementioned performance evaluation metrics.

TABLE II. COMPARING THE PERFORMANCE OF MACHINE LEARNING CLASSIFIERS

	Logistic Regression	*Random Forest*	*KNN*	*Gradient Boost*	*SVC*	*Decision Tree*	*LDA*	*Gaussian Naive Bayes*	*XGBoost*
Accuracy	0.781167	0.8148	0.756	0.8198	0.7812	0.7295	0.809667	0.3815	0.820833
Recall	0.781167	0.8148	0.756	0.8198	0.7812	0.7295	0.809667	0.3815	0.820833
Precision	0.610221	0.7955	0.7086	0.8020	0.6102	0.7368	0.789711	0.741742	0.803417
F1	0.685193	0.7948	0.7218	0.7978	0.6852	0.733	0.773427	0.379686	0.798653

The results infer that the best model can be developed using the XGBoost classifier among all the classifiers that we have used.

D. Performance of the Artificial Neural Network Model

The ANN model developed consisted of an input layer and 4 hidden layers with 'Relu' as the activation. Relu was used as it helps in solving the vanishing gradient problem. The output layer returned a single value as it had 'sigmoid' as the activation. With an accuracy of only 78% and an F1-score of 68%, we discarded the artificial neural network model for poor performance in this case. At the same time, we got a better result without tuning the XGBoost algorithm and believed it is wiser to tune XGBoost further as compared to ANN.

E. Hyper-parameter tuning of the best model

As we can observe from the above table, XGBoost gives us the best accuracy and F1-score as compared to all the models. Thus, we performed hyper-parameter tuning on the same using grid search and cross-validation which took a dictionary of parameters and iterated each list of parameters through different combinations of the model using cross-validation. Evaluation using cross-validation gave us the best value for each parameter based on the value of squared error.

TABLE III. PARAMETRIC CHANGES OF TUNING OF THE XGBOOST MODEL

Sr. No.	*Variable Name*	*Objective*	*Default Value*	*Best Value*
1	colsample_bytree	Fraction of columns to be randomly sampled for each tree	1.0	0.7
2	min_child_weight	The minimum sum of weights of all observations required in a child	1	5
3	subsample	The fraction of observations to be randomly sampled for each tree	1	0.5
4	n_estimators	Number of estimation rounds	100	200

After tuning, the model provided us with an accuracy of 82.23%.

VI. CONCLUSION AND FUTURE SCOPE

After a thorough comparison of various machine learning classifiers based on their performance on a Taiwanese credit risk dataset evaluated using 4 metrics, it is observed that the difference in error between all ensemble decision tree algorithms was very less and the best performing model was developed using the XGBoost classifier. However, the second-lowest performing classifier was the traditional decision tree algorithm. The least accuracy was observed when the Gaussian Naive Bayes model was tested. Moreover, the ANN model was discarded because of its poor performance as compared to that of the tree algorithms.

After hyper-parameter tuning of the best performing classifier, we computed the list of features that proved to be the most significant for building the XGBoost model. The most important features were the amount of payment due for the previous month and the limit balance allotted to the customer. We also observed that certain features like marital status and age hardly make any difference on whether the customer will default his payment or not.

It was also observed that the data had sufficient information about non-defaulters, but lacked to train the model efficiently to predict defaulters. This was indicated by a low recall score for all predictions. As a result, a relatively higher number of false negatives were observed. These errors definitely have a high negative impact as they mispredict non-credible customers as credible and may lead to excessive losses to the banks. In order to receive better results, we would require a dataset that gives adequate information about defaulters to analyse their behaviour. Thus, this research enables researchers to use larger datasets in future and use advanced classifiers to achieve better results which would aid the financial industry.

ACKNOWLEDGMENT

We would like to express immense gratitude to Prof. Pranav Nerurkar for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time and effort so generously towards our development in the field of research is much appreciated.

REFERENCES

[1] Trevor Hunnicutt, Past-due student loans, credit card debt could weigh on U.S. growth, 2019. Retrieved from Reuters, [Online]. Available: https://www.reuters.com/article/us-usa-economy-debt/past-due-student-loans-credit-card-debt-could-weigh-on-u-s-growth-idUSKCN1V31HA [Accessed: 04-Apr-2021].

[2] Makram Soui, Salima Smiti, Salma Bribech, Ines Gasmi, Credit Card Default Prediction as a Classification Problem, (https://link.springer.com/chapter/10.1007/978-3-319-92058-0_9) [Accessed: 09-Apr-2021].

[3] The Impact of the Economic Crisis on East Asia: Policy Responses from Four Economies. United Kingdom: Edward Elgar Publishing Limited, 2011.

[4] Recent Episodes of Credit Card Distress in Asia, BIS Quarterly Review, June 2007

[5] Taiwan card payments to rise 5.6% in 2020 despite COVID-19, reveals GlobalData, Global Data, December 2020, https://www.globaldata.com/taiwan-card-payments-rise-5-6-2020-despite-covid-19-reveals-globaldata/ [Accessed: 19-May-2021].

[6] I-Cheng Yeh, Che-hui Lien, The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients, Expert Systems with Applications, Volume 36, Issue 2, Part 1, 2009, Pages 2473-2480, ISSN 0957-4174, (https://www.sciencedirect.com/science/article/pii/S0957417407006719) [Accessed: 04-Apr-2021].

[7] Pandey, T. N., Jagadev, A. K., Mohapatra, S. K., & Dehuri, S. (2017). Credit risk analysis using machine learning classifiers. 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS). https://doi.org/10.1109/icecds.2017.8389769

[8] Addo, P. M., Guegan, D., & Hassani, B. (2018). Credit risk analysis using machine and deep learning models. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3155047

[9] Charleonnan, A. (2016). Credit card fraud detection using RUS and MRN algorithms. 2016 Management and Innovation Technology International Conference (MITicon). https://doi.org/10.1109/miticon.2016.8025244

[10] A. Petropoulos, V. Siakoulis, E. Stavroulakis, and A. Klamargias, "A robust machine learning approach for credit risk analysis of large loan level datasets using deep learning and extreme gradient boosting," use big data Anal. Artif. Intell. Cent. Bank., vol. 50, no. August, pp. 30–31, 2018, [Online]. Available: https://www.bis.org/ifc/publ/ifcb49_49.pdf.

[11] Andrés Alonso and José Manuel Carbó, Machine Learning in Credit Risk: Measuring the dilemma between prediction and supervisory cost, 2020

[12] Mit.edu. [Online]. Available: https://ocw.mit.edu/courses/health-sciences-and-technology/hst-951j-medical-decision-support-spring-2003/lecture-notes/lecture7_8.pdf. [Accessed: 04-Apr-2021].

[13] E. Goel, Computer Science & Engineering &GZSCCET Bhatinda, Punjab, India, E. Abhilasha, and Computer Science & Engineering &GZSCCET Bhatinda, Punjab, India, "Random Forest: A Review," Int. j. adv. res. comput. sci. softw. eng., vol. 7, no. 1, pp. 251–257, 2017.

[14] K.-N. Neighbors, "15.097 MIT, Spring 2012, Cynthia Rudin Credit: Seyda Ertekin," Mit.edu. [Online]. Available: https://ocw.mit.edu/courses/sloan-school-of-management/15-097-prediction-machine-learning-and-statistics-spring-2012/lecture-notes/MIT15_097S12_lec06.pdf. [Accessed: 04-Apr-2021].

[15] J. M. Islam, J. Q. M. Wu, and M. Ahmadi, "Investigating The Performance Of Naïve-Baysed and K- Nearest Neighbor Classifiers'," 2007, pp. 1541–1546.

[16] V. Jakkula, "Tutorial on Support Vector Machine (SVM)," Neu.edu. [Online]. Available: https://course.ccs.neu.edu/cs5100f11/resources/jakkula.pdf. [Accessed: 04-Apr-2021].

[17] P. Danenas, G. Garsva, and S. Gudas, "Credit risk evaluation model development using support vector based classifiers," Procedia Comput. Sci., vol. 4, pp. 1699–1707, 2011.

[18] X. Zeng, S. Yuan, Y. Li, and Q. Zou, "Decision tree classification model for popularity forecast of Chinese colleges," J. Appl. Math., vol. 2014, pp. 1–7, 2014.

[19] Max Welling, "Fisher Linear Discriminant Analysis", https://www.ics.uci.edu/. [Online]. Available at: https://www.ics.uci.edu/~welling/teaching/273ASpring09/Fisher-LDA.pdf. [Accessed: 05-Apr-2021].

[20] B. M. Gayathri, C. P. Sumathi, "An Automated Technique using Gaussian Naïve Bayes Classifier to Classify Breast Cancer", Available at: https://www.ijcaonline.org/archives/volume148/number6/gayathri-2016-ijca-911146.pdf [Accessed: 05-Apr-2021].

[21] Cheng Li, "A Gentle Introduction to Gradient Boosting", [online] Available at: <http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf> [Accessed 4 April 2021].

[22] M. S. Mhatre, F. Siddiqui, M. Dongre, and P. Thakur, "Prediction Technique," Ijser.org. [Online]. Available: https://www.ijser.org/researchpaper/A-Review-paper-on-Artificial-Neural-Network--A-Prediction-Technique.pdf. [Accessed: 04-Apr-2021].

[23] E. P. Kumar and E. P. Sharma, "Artificial Neural Networks-A Study," Ijeert.org. [Online]. Available: http://www.ijeert.org/pdf/v2-i2/24.pdf. [Accessed: 04-Apr-2021].

Prediction of Stroke possibilities using various Classification Models

Kameshwaran Ganesan, B.E.,
Chennai, Tamil Nadu.
kameshwaran.ganesan56@gmail.com

Pavithra Mamallan, B.Tech., MBA.,
Chennai, Tamil Nadu.
pavi.illa@gmail.com

Abstract—This problem concerns with **predicting** whether the patient will get stroke in the future with predictors like age, gender, smoking status, body mass index, whether they had heart disease, whether they had hypertension etc. Since the output variable is **categorical** in nature, it is a **classification problem**. Many classification techniques are used with the help of three main Business Analytics tools such as **Excel, R and Python**. The data is understood through **Exploratory Data Analysis**, then Data Pre-Processing is done to prepare the data, various models are built on the data and finally Error metrics are used to compare the results.

1. Introduction

1.1 Introduction to Health Care Industry

Healthcare has become one of India's largest sectors - both in terms of revenue and employment. Healthcare comprises hospitals, medical devices, clinical trials, outsourcing, telemedicine, medical tourism, health insurance and medical equipment. The Indian healthcare sector is growing at a brisk pace due to its strengthening coverage [2], services and increasing expenditure by public as well private players.

Indian healthcare delivery system is categorized into two major components - public and private. The Government, i.e., public healthcare system comprises limited secondary and tertiary care institutions in key cities and focuses on providing basic healthcare facilities inthe form of primary healthcare centers (PHCs) in rural areas [1]. The private sector provides majority of secondary, tertiary and quaternary care institutions with a major concentration in metros, tier I and tier II cities.

1.2 Stroke

Stroke is a disease that affects the arteries leading to and within the brain. It is the No. 5cause of death and a leading cause of disability in the United States. A stroke occurs when a blood vessel that carries oxygen and nutrients to the brain is either blocked by a clot or bursts (or ruptures). When that happens, part of the brain cannot get the blood (and oxygen) it needs, so it and brain cells die [3].

1.3 Problem Statement

You have dataset consisting of the past history of the patients suffering from Stroke. Youhave to predict whether the new patients will be affected by Stoke based on the given attributes.

1.4 Data

The dataset "train_strokes" contains 43400 observations and 12 variables. The dependent variable is 'stroke.'

- Source of data – Kaggle.

	Dtype
id	int64
gender	object
age	float64
hypertension	int64
heart_disease	int64
ever_married	object
work_type	object
Residence_type	object
avg_glucose_level	float64
bmi	float64
smoking_status	object
stroke	int64

Output Variable: Categorical
stroke-0 - no stroke, 1 - suffered stroke

2 Data Preparation

Data preparation (also referred to as "data pre-processing") is the process of transforming rawdata so that data scientists and analysts can run it through machine learning algorithms to uncover insights or make predictions.

Most machine learning algorithms require data to be formatted in a very specific way, so datasets generally require some amount of preparation before they can yield useful insights. Some datasets have values that are missing, invalid, or otherwise difficult for an algorithm to process. If data is missing, the algorithm can't use it. If data is invalid, the algorithm produces less accurate or even misleading outcomes. Some datasets are relatively clean but need to be shaped (e.g., aggregated or pivoted) and many datasets are just lacking useful business context (e.g., poorly defined ID values), hence the need for feature enrichment [4]. Good data preparation produces clean and well-curated data which leads to more practical, accurate model outcomes.

2.1 Missing Value Analysis

Any predictive modeling requires an important step which looks at the data deeply before we start modeling. However, in data mining terms **looking at data** refers to so much more than just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as **Exploratory Data Analysis**.

	No of Missing Values
id	0
gender	0
age	0
hypertension	0
heart_disease	0
ever_married	0
work_type	0
Residence_type	0
avg_glucose_level	0
bmi	1462
smoking_status	13292

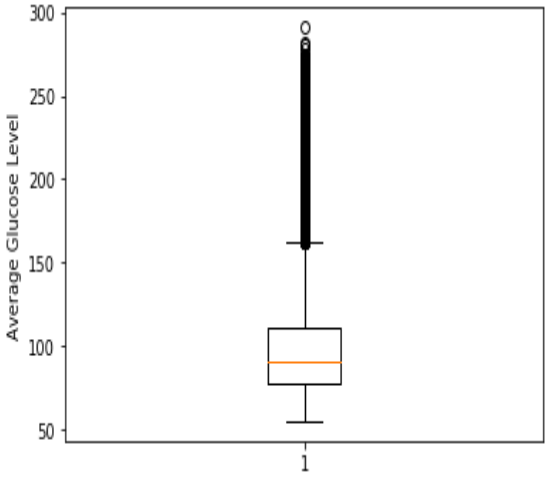
	No of Missing Values
stroke	0

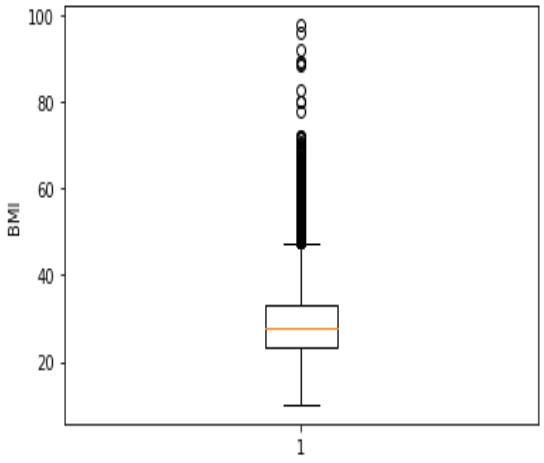
- Replacing na with new feature "Not disclosed"
  Code: data['smoking_status'].fillna("Not disclosed",inplace=True)
- Since the value of BMI is less than 3% of the whole data, we are dropping the 1462 Null values in the dataset.

2.2 Outlier Analysis

Outliers are extreme values that deviate from other observations on data; they may indicate variability in a measurement, experimental errors or a novelty. In other words, an outlier is anobservation that diverges from an overall pattern on a sample.

The plots clearly show that the variables contain extreme values or outliers. The numeric variables are selected and plotted using box plot. In this case, the outliers are crucial for the analysis part and hence it is not removed.





2.3 Feature Selection

$$X_{new} = \frac{X_i - X_{mean}}{\text{Standard Deviation}}$$

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of the model. The data features that are used to train the machine learning models have a huge influence on the performance. Irrelevant or partially relevant features can negatively impact model performance.

The variable 'id' is not important for the analysis. So, this variable is removed. All the categorical variables are selected and dummy variables are created for them.

Dummy variables are created for 'gender', 'ever_married', 'work_type', 'Residence_type', 'smoking_status'.

Dummy variables are useful because they enable us to use a single regression equation to represent multiple groups. This means that we don't need to write out separate equation models for each subgroup. The dummy variables act like **'switches'** that turn various parameters on and off in an equation.

2.4 Feature Scaling

Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units [3]. If feature scaling is not done, then a machine learning algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values.

Techniques to perform Feature Scaling

Consider the two most important ones:

- **Min-Max Normalization:** This technique re-scales a feature or observation valuewith distribution value between 0 and 1.

- **Standardization:** It is a very effective technique which re-scales a feature value sothat it has distribution with 0 mean value and variance equals to 1.

3 Modeling

In the previous sections we have done all the pre-processing steps in the dataset to developthe model. Now, our problem statement is to predict the stroke, whether the patient will get stroke in the future or not. The target variable is categorical in nature and so we build models for Classification analysis. Always, we have move from simple to complex. Hence, the first model that we are going to build is Logistic Regression.

In this dataset, the continuous variables are selected and are normalized. The variables 'age', 'avg_glucose_level', 'bmi' are normalized.

2.5 Label Encoding

Label Encoding refers to converting the labels into a numeric form so as to convert them into the machine-readable form. Machine learning algorithms can then decide in a better way how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning

2.6 Sampling

Sampling is the selection of a subset (a statistical sample) of individuals from within a statistical population to estimate characteristics of the whole population.

- **Population parameter**. A population parameter is the true value of a population attribute.
- **Sample statistic**. A sample statistic is an estimate, based on sample data, of a population parameter.

For this dataset, simple random sampling is used since the dependent variable is continuous. The dataset is divided into train and test data. 80% of the data is separated for training thedata and the remaining 20% is for testing the data.

The process of training an ML model involves providing an ML algorithm (that is, the learning algorithm) with training data to learn from. The term ML model refers to the model artefact that is created by the training process. The training data must contain the correct answer, which is known as a target or target attribute. The learning algorithm finds **patterns in the training data** that map the input data attributes to the target (the answer that you want to predict), and it outputs an ML model that captures these patterns.

$$\log(P(y=1) / 1-P(y=1))=\log(P(y=1) / P(y=0))=\beta 0+\beta 1x1+\ldots+\beta pxp$$

$$X_{new} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

And then we move on to complex algorithms, Naïve Bayes, Decision Tree.

3.1 Logistic Regression

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the

relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables [4]. At the centre of the logistic regression analysisis the task estimating the log odds of an event. Mathematically, logistic regression estimates a multiple linear regression function defined as:

3.2 Naïve Bayes Classification

The Naive Bayes Classifier technique is based on the so-called Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Despite its simplicity, NaiveBayes can often outperform more sophisticated classification methods. Naive Bayes classifiers can handle an arbitrary number of independent variables whether continuous or categorical. Given a set of variables, X = {x1,x2,x...,xd}, we want to construct the posterior probability for the event Cj among a set of possible outcomes C = {c1,c2,c...,cd}[5]. In a more familiar language, X is the predictors and C is the set of categorical levels present in the dependent variable. Using Bayes' rule:

where p(Cj | x1,x2,x...,xd) is the posterior probability of class membership, i.e., the probability that X belongs

$$p(C_j \mid x_1, x_2, \ldots, x_d) \propto p(x_1, x_2, \ldots, x_d \mid C_j) p(C_j)$$

to Cj. Since Naive Bayes assumes that the conditional probabilities of the independent variables are statistically independent, we can decompose the

$$p(X \mid C_j) \propto \prod_{k=1}^{d} p(x_k \mid C_j)$$

likelihood to a product of terms:

3.3 Decision Tree Classification

Decision tree classifiers are utilized as a well-known classification technique in different pattern recognition issues, for example, image classification and character recognition. Decision tree classifiers perform more successfully, specifically for complex classification problems, due to their high adaptability and computationally effective features. Besides, decision tree classifiers exceed expectations over numerous typical supervised classification methods [6].

In particular, no distribution assumption is needed by decision tree classifiers regarding the input data. This particular feature gives to the Decision Tree Classifiers a higher adaptability to deal with different datasets, whether numeric or categorical, even with missing data. Also, decision tree classifiers are basically nonparametric [6]. Also, decision trees are ideal for dealing with nonlinear relations among features and classes. At long last, the classification procedure through a tree-like structure is constantly natural and interpretable.

In the event that each parent hub is part into two descendants, the decision tree is frequently known as a binary tree, and the inherent decision rule can be communicated as a dyadic Boolean operator with the end goal that the data points focuses are split based on condition rules satisfaction.

'rpart' function is used for Decision Tree Regression Analysis. rpart() function helpsestablish a relationship between a dependant and independent variables so that a business can understand the variance in the dependant variables based on the independent variables. '**Tree.DecisionTreeClassifier**' [6] is the function used in Python.

4 Error Metrics

Predictive Modeling works on constructive feedback principle. You build a model. Get feedback from metrics, make improvements and continue until you achieve a desirable accuracy. Evaluation metrics explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results. Simply, building a predictive model is not your motive. But, creating and selecting a model which gives high accuracy on out of sample data. Hence, it is crucial to check accuracy of the model prior to computing predicted values.

There are several ways to evaluate the model. In this project, I have used Confusion Matrix, Accuracy, Precision, and False Negative Rate

NB_Predictions / Stroke	0	1
0	9386	925
1	113	61

4.1 Confusion Matrix

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

Log_Predictions / Stroke	0	1
0	10309	2
1	174	0

DT_Predictions / Stroke	0	1
0	10150	161
1	160	14

4,2 Accuracy and PrecisionAccuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

Results:

Almost all the models are performing well with respect to Accuracy.

Models	Accuracy
Logistic Regression	98.23%
Naïve Bayes	90.10%
Decision Tree	96.96%

Precision

Precision attempts to answer the following question: "What proportion of positive identifications was actually correct?"

Precision is defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP}$$

The Precision value for Logistic Regression is 99.98% and that of Decision Tree is 98.44%.

Models	Precision
Logistic Regression	98.38%
Naïve Bayes	98.03%
Decision Tree	98.44%

False Negative Rate

The **false negative rate** is the proportion of positives which yield **negative** test outcomes with the test, i.e., the conditional probability of a **negative** test result given that the condition being looked for is present.

From the False Negative Rates, the **Logistic Regression model** is selected as the best fitsince it contains lowest FNR. This means that the model is reliable to a greater extent than other models.

False negative rate (β) = type II error
= 1 − sensitivity
= FN / (TP + FN)

Models	FNR
Logistic Regression	**0.19%**
Naïve Bayes	8.97%
Decision Tree	1.56%

5 Conclusion

From all the Error metrics, it is arrived at a conclusion that Logistic Regression is more suitable for this dataset than Decision Tree Regression and Naïve Bayes. All the models are performing better with respect to Accuracy and Precision. Since this dataset is healthcare dataset, False Negative Rate plays a crucial part of analysis. Patients should be precisely predicted whether they have the chance of getting stroke or not. If not, it will create a big issue. Patients who have been detected early can be saved easily. Hence, the model with lowest FNR is chosen for prediction.

Hence, Logistic Regression model is used for predicting the test dataset. Before giving the dataset into the model, it is subjected to all the pre-processing techniques.

Accuracy of Logistic Regression is 98.23%Precision of Logistic Regression is 98.38%
False Negative Rate of Logistic Regression is 0.19%

References

1. Michael J Ward, Keith A Marsolo, Craig M Froehle, Applications of Business Analytics in Healthcare, *Bus Horiz,* 2014, 57(5), 571-582.
2. Setkowski K, Mokkenstorm J, van Balkom AJ, et alFeasibility and impact of data-driven learning within the suicide prevention action network of thirteen specialist mental healthcare institutions (SUPRANET Care) in the Netherlands: a study protocolBMJ Open 2018;8:e024398. doi: 10.1136/bmjopen-2018-024398
3. Salazar-Reyna, R., Gonzalez-Aleu, F., Granda-Gutierrez, E.M., Diaz-Ramirez, J., Garza-Reyes, J.A. and Kumar, A., 2020. A systematic literature review of data science, data analytics and machine learning applied to healthcare engineering systems. Management Decision, pp. 1-20.
4. Sandro Sperandei, Understanding logistic regression analysis, *Biochemia Medica*, 2013, 24(1), 12-8
5. Chao-Ying Joanne Peng et al, An Introduction to Logistic Regression Analysis and Reporting, *The Journal of Educational Research,* 2002, 96(1).
6. Nora Galambos, What Satisfies Students? Mining Student-Opinion Data with Regression and Decision Tree Analysis, *Academia,* 2004, 45(3), 251-269.
7. https://www.mastersindatascience.org/resources/what-is-business-analytics/
8. https://www.stroke.org/en/about-stroke/stroke-symptoms
9. https://www.ibef.org/industry/healthcare-india.aspx
10. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4242091/
11. https://www.datapine.com/healthcare-analytics
12. https://www.dummies.com/programming/big-data/phase-1-of-the-crisp-dm-process-model-business-understanding/
13. https://www.healthline.com/health/stroke-types#symptoms
14. https://medium.com/greyatom/why-how-and-when-to-scale-your-features- 4b30ab09db5e
15. https://www.geeksforgeeks.org/ml-feature-scaling-part-1/
16. https://www.sciencedirect.com/topics/computer-science/decision-tree-classifier

AI Powered Forecasting for Workforce Management

K.A.V.Lakshmi Raghavan
Data Scientist, Genpact
kumandurianantthavenkata.lakshmiraghavan@genpact.digital

Yogita Rani
Manager, Genpact
yogita.rani@genpact.digital

Ladle Patel
Senior Manager, Genpact
ladle.patel@genpact.digital

Rajeev Ranjan
Assistant Vice-President, Genpact
rajeev.ranjann@genpact.digital

***Abstract*— In this paper, we present our approach on using time series forecasting for managing resource allocation for one of the Social Media UPI. With the rapid increase in online payments, the business is facing challenges in resource allocation across three vendors to monitor its payment requests categorized under seven hierarchical processes labelled as Towers. Each of these Towers is further divided into various sub-processes called Products and then Alert Queues. Since the workforce allocation depends upon the incoming volume requests for these processes, it is vital to have a proper volume forecasting system to plan for the optimal resource allocation in advance. We forecasted volumes for all Products and Alert Queues across all the Towers and categorized them under three different risk buckets depending upon the Mean Absolute Percentage Error (MAPE) observed on test data. Using the forecasted volumes and the risk category the Product falls under, capacity planning for the future dates is made easier and the client was able to reduce its resources by 26%.**

Keywords— Time-series forecasting, Tower, Product, Alert Queue, Workforce Management, Business input, Risk category, Service Level Agreement, Manual Inflow count

I. INTRODUCTION

In this paper, we discuss how we integrated workforce and Time Series forecasting for Social Media's United Payment Interface. We provide a brief overview of the recurring terms in the paper below

A. Time-series Forecasting

A time series is a sequence of time-ordered variables that measures some process. It is present in many sectors, for instance, sales data or demand data of a product, sensor measurements, stock market, etc. Forecasting is a technique that uses past data to make optimal business decisions that determine future trends. It has applications in all sectors such as sales and demand forecasting, weather forecasting, stock markets, and intrinsic applications such as capacity planning for optimal resource allocation

B. Workforce Management

Workforce management is the process of optimal allocation of resources to specific tasks to maximize the organization's performance. Efficient workforce planning is essential to advance the productivity and competency of an organization and to run a cost-efficient business.

C. United Payment Interface

United Payment Interface (UPI) is a system that allows users to merge multiple bank accounts into a single mobile application to make instantaneous and seamless P2P (peer to peer) and P2M (peer to market) transactions. Multiple processes like KYC happen in the background of the UPI to facilitate transactions. These KYC procedures fall under the Anti-Money Laundering (AML) policies of the organization. The financial Institutions supporting the UPI demand the customer to provide Due Diligence Information and keep tabs on Politically Exposed Persons and transactions happening to Sanctioned nations. More details of these processes are explained in section-2 of the paper.

D. Service Level Agreement

Service Level Agreement (SLA) is a commitment between the client and the vendor that covers various business aspects such as quality of service expected by the vendor, various performance metrics, etc. One of such aspects covered in the SLA of this project is the time taken by an FTE to work on a manually closed ticket. This time varies from Tower to Tower ranging from 24 hours for Sanctions to 72 hours for DD. Throughout the rest of this paper, the term SLA refers to this time.

E. This paper,

discusses how we integrated time series forecasting techniques to facilitate workforce management for one of the Social Media's UPI. We also discuss the overall business structure, statistical forecasting techniques and ensemble methods we used, evaluation metrics, and how the organization benefitted from our forecasting.

F. Paper Structure

In Section 2 we discuss Business structure in more detail. Section 3 describes the forecasting techniques we used and how we integrated business inputs into our forecasts. In Section 4 we provide our model, and we summarize the impact of our work in Section-5.

II. Business Structure

A. Basic Process Overview

In this paper, we are focusing on Workforce Management for an online UPI platform. Multiple processes happen in this UPI's background to facilitate safe transactions and keep tabs on fraudulent transactions and businesses. These processes are called Towers. For this specific UPI, there are seven towers as Identity Verification, Sanctions, Due Diligence, Transaction Monitoring, Politically Exposed Persons, etc. Each of these towers has several Products, and each of these Products has varying escalation levels called Alert Queues. The Products are unique for all the Towers, and the Alert Queue hierarchy is the same for all the products in a Tower and may vary from Tower to Tower

Whenever a transaction or a request from the customer happens in the UPI, a ticket is being raised under a specific Product depending upon the category of the transaction. Most of these tickets are handled by the Artificial Intelligence system developed by the UPI organization. However, manual intervention is required for some of them. These are called Manually closed tickets, Manual Inflow counts, or Manual Inflow volumes. These tickets are initially categorized under base level escalation Alert Queue (L0 or L1 depending upon which Tower the Product falls under). Then, depending upon the criticality of the Product, it is escalated to higher Alert Queues. The organization has outsourced the task of working on these manually closed tickets to 5 vendors. Each vendor works on different towers and escalation queues. Each vendor plan to hire the required workforce two months in advance. These manually closed tickets are highly volatile across most of the towers. If not adequately planned, it may lead to overstaffing, which leads to organizational cost, or understaffing which may lead to a drop in organizational performance. Hence it is essential to have a proper forecasting system in place to maximize organizational performance

B. WFM Process Overview

The entire WFM structure in this project goes through multiple processes, with each dealing with specific tasks as shown in Fig 1. The first one is forecasting. To hire an optimal workforce, we started with forecasting the manually closed tickets. This forecasting is done across all Towers, Products, and Alert Queues. Depending upon the MAPE of our forecasts and the volatile nature of the Products, they are divided into 3 risk buckets. Once these tickets are forecasted, these numbers are converted into Full-Time Employees (FTEs) by the capacity planning team depending upon which risk bucket the Product falls under. Based on the FTE count, agents are aligned to work on the tickets of the Products and their corresponding escalation levels. Once, we have agents and tickets, the scheduling team schedules all these tickets against agents. Real-time monitoring keeps track of whether there is any sudden change or a sudden product that needs more focus, based on this they do real-time reassigning. For reporting, we maintain a daily tracker which compares day-to-day forecasts and actuals. If there is any sudden surge in actuals then we connect with the business team to find out the reason if this is because of backlogs, breakdowns, or if there is any change in the business model for that Product. Based on this input we either impute the data or incorporate this business knowledge in the forecasting for the next iteration.

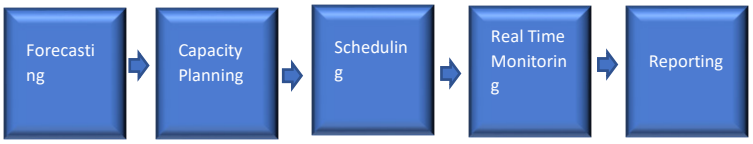


Fig. 1: WFM Process Overview

III. Forecasting

A. Forecasting Process Overview

The forecasting of Manual Inflow counts is done for 7 Towers. Five of these towers are Sanctions, Due Diligence (DD), Identity Verification (IDV), Transaction Monitoring (TM), Politically Exposed Persons (PEP). The remaining two towers cannot be named due to the company's policy. The Sanctions tower accounts for approximately 70% of the volume. It has 74 Products. For each Product, the base escalation Alert Queue is 'L0'. From then on it escalates to 'L1A/L1B' then to 'L2A' and finally to 'L3'. This escalation hierarchy is not the same for all the Towers. For example, in Due Diligence, the escalation hierarchy is 'L1' to 'L2' and then to 'L3'. In all towers, the highest escalation is 'L3' and its numbers are very few. Hence, we focus more on other escalation queues with most of our discussion based on the basic escalation queue ('L0' or 'L1' depending upon the tower). The basic escalation queues constitute more than 80% of the volumes in all the towers. The Process of forecasting for each Product and Alert Queue combination is shown in Fig 2.

Fig. 2: Forecasting Process Overview

In the coming paragraphs, we explain how we forecasted for the basic escalation Alert Queues and

how we extended our forecasts to higher escalation Alert Queues.

B. Data Overview

Whenever a new product is launched into the market, it is initially released in specific regions or countries. Depending upon its performance, the decision is made whether to scale up the Product release to other regions or not. This scaling up may happen in multiple phases. When we started the forecasting process, 99% of the Products were in the final phase, which means they were released at full scale. For all these Products, we have, at maximum, only nine months of data. For some of the Products, we have even fewer data. This being the case, we don't have any annual seasonal pattern in the data. Also, we cannot rely on the data of initial phases because the absolute Manual inflow count numbers and the weekly patterns are entirely different from one phase to another. Even the available nine months of data is not entirely reliable because tool breakage issues were. So, the recorded numbers in this period will be lower than usual. Once the tool is repaired, the old tickets are dumped into the current period. These are called backlogs. In these cases, the actual numbers are less than that of recorded numbers. So, we impute these numbers by discussions with the business and observations from the graphs. Occasionally, a new Product may be added to the Tower during the process of forecasting. We had to rely more on the business inputs for products like these.

C. Data Imputation

Multiple imputation strategies were used after observing the graphs of each Product and business inputs from the client.

1. Backlogs in Sanctions and PEP are highly common. For example, the whole data in March and April of the year 2021 have backlogs for both these towers. To impute these backlogs, we considered the data from Jan'21 to July'21 for all the Products under these Towers, calculated the mean Manual Inflow count in this period, and through various experimentations, found out that actual numbers are to be likely around 10% of this mean. So, we replaced every number that is more than 10% of the mean with the mean value of that Product and Alert Queue combination. The experimentations for arriving at this 10% threshold are based upon various discussions with the business, observation of historical data. Other Towers don't have backlogs.

2. We observed two weeks of missing data for one of the major volume contributor Products in the Sanctions tower. For this Product, we replaced the missing values based on the data two weeks before and two weeks after this missing period and the weekly pattern we observed in these four weeks.

3. Additionally, in all the towers, a sudden peak for a day or two and a sudden decrease in a day or two may be observed for a Product due to backlogs. These numbers are imputed based on the mean Manual Inflow count observed for that Product in that month. The imputation is done based on a certain threshold such that these peaks and troughs fall outside the bounds of these thresholds. These thresholds vary from tower to tower and are decided by observing the data of that Tower.

These steps constitute data imputation. This data can now be used for forecasting.

D. Forecasting

Each Tower has multiple Products, and each Product follows a unique business structure. Sanctions Tower has 74 Products, and DD has 21 Products, etc. The forecasting for each Tower is done separately. In an ideal scenario, each Product is to be forecasted independently. However, in this case, we have many Products, and for each Product, we have multiple Alert Queues. This gives around 204 different combinations in Sanctions alone. So, forecasting for each variety is very inefficient. We observed that a few Products contribute to 80% of the Volume in each Tower. So, we predicted for these top 80% volume contributing Products in each Tower separately, and for remaining combinations, we combine the volumes as one, do the forecast, and divide the forecasted volumes depending on the recent 60 days of historical contribution.

The forecasting methods we used are the average of last 15 days, Simple Exponential Smoothening (SES), Double Exponential Smoothening (DES), ARIMA, ARIMAX[1], Weighted Average models, Prophet [2], TBATS [3], and other ARIMAX ensembles which will be discussed in detail in the later sections.

The capacity planning is done based on the weekly volume numbers. So, it is essential to capture the weekly patterns in our forecasts. We have observed that the ARIMAX model with the week number of the month as a regressor variable perfectly captures the weekly trends. On the other hand, TBATS and Prophet models, though highly sophisticated, did not give good results even with hyperparameter tuning due to less available data. So, for all the top 80% of the Products across all the Towers, we have used different variants of ARIMAX models and their ensembles.

1. For Sanctions, we used three different regressor variables. One is Week number in a month. The First seven days correspond to the first week, the next seven days as the second week, the next seven as the third week, and all the remaining days are assigned to the fourth week. We inputted this variable into the model through one-hot encoding. The next one is the day of the week. Again, we one-hot encoded this variable and inputted it into the model. The final regressor variable that we used is the trend. It is a scalar that ranges from 0 to the length of the training data. We used two ARIMAX variants. One variant uses both the week of the month and trend regressors, and the other variant uses only the

week of the month regressor. For Sanctions, we used the average value of these two variants for forecasting

2. For PEP, we used an ensemble of two ARIMAX variants. However, this ensemble is different from the one we used for the Sanctions. The first variant uses the day of the week and the day of the month as regressor variables. The second variant uses the month of the year, trend regressors, and the regressors mentioned in the first variant. We have observed that we are getting good MAPE on our forecasts at a monthly level using the second variant. However, the second variant is not capturing the weekly pattern. The first variant captures the weekly pattern perfectly, but the forecasts are not accurate. So, we decided to capture the pattern from the first variant and the monthly number from the second variant. We did this by scaling the forecasts of the first variant to match its monthly forecast number with the second variant. This ensemble method worked perfectly for PEP.

3. For DD, TM, and IDV we achieved good MAPE on our forecasts using a single ARIMAX model that uses the day of the week and the week of the month as regressor variables.

4. Similar methods are applied for the remaining two Towers.

5. For the bottom 20% of the Products across each tower, we used the forecasting method that gave us the best MAPE on the last month of the training data.

6. We applied similar forecasting methods to only some higher Alert Queues (Alert Queues other than L0 or L1) depending upon their volatility. These techniques worked well for Products with low volatility. For Products with high volatility, forecasts using the average one-month escalation rates from L0 or L1 to higher Alert Queues gave the best results.

E. Business Inputs

Many external factors such as holiday effects, promotional events, etc., affect the Manual inflow count of the Products. One of such critical external effects is the Giving Tuesday holiday. However, it is impossible to capture the holiday's effect using the forecasting methods with only nine months of available data. So, we made different observations for each Tower on the limited, pre-final phase data we had. These observations include how many weeks before Giving Tuesday has the effect of the holiday started and how many weeks it lasted. This period varies from 5 weeks to 2 weeks depending upon the Tower. Another observation is the percentage changes in the overall Volume of the Tower during this period. We validated our observations with the client and used them in the forecast as follows. Instead of using model-produced numbers during this period, we used the percentage changes from the previous weeks. One more external factor is the promotional events. For promotional events, the extra volumes are added on top of the forecasted volumes depending upon the promotion type and the promotion scale

IV. Experiments and Results

A. Model Evaluation Metrics

We used MAPE (Mean Absolute Percentage Error) on one month of testing data to make model evaluations.

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{A_t - F_t}{A_t}\right| \quad (1)$$

A_t is the Actual volume at time t

F_t is the Forecasted volume at time t

N is the length of test data

Instead of directly using (1) for calculating MAPE, we decided to incorporate the SLA information of each Tower. Instead of taking the Actual and Forecast numbers for each day, we took a cumulative sum of the rolling windows of the Actual and the Forecasted numbers with a window length equal to that of SLA. For example, DD has an SLA of 72 hours. While calculating MAPE for DD, for each observation of the Actual, we took a cumulative sum of 3 days rolling window instead of a single day number. The same logic is applied for the forecasted volumes as well. So, we used the following formula for MAPE

$$MAPE = \frac{1}{N-SLA}\sum_{t=1}^{N-SLA}\left|\frac{\sum_{t'=t}^{SLA+t} A_{t'} - \sum_{t'=t}^{SLA+t} F_{t'}}{\sum_{t'=t}^{SLA+t} A_{t'}}\right| \quad (2)$$

SLA is the SLA of the Tower the Product falls under

B. Results

Using (2), we calculated the MAPE for all the Products across all the Towers and categorized these Products under 3 risk buckets. These buckets are summarized in Table-1.

Table-1: Percentage of Products and Alert Queue combinations across different risk buckets

Risk Bucket	**MAPE**	**Percentage of Product and Alert Queue combinations in this risk bucket**
Category -1	$MAPE \leq 10\%$	12%
Category -2	$10\% < MAPE \leq 20\%$	64%
Category -3	$MAPE > 20\%$	24%

As seen from Table-1, the majority of the Products which also contribute to the majority of the volumes are under Category-2. These Products require a little buffer. Category-3 Products are highly volatile which often require more information from the business to plan for the resources.

V. Conclusion

A. Summary

This paper showed how we incorporated forecasting methods into workforce management. Due to limited availability and the volatile nature of the data, various business inputs are incorporated into these methods, especially for dealing with backlogs, holiday effects, and promotional events. We used MAPE as a model evaluation metric since it penalizes both over and under forecasting. The usage of the cumulative sum of rolling windows of SLA length is more sensible because tickets are worked depending upon their respective SLAs. Thus, we can provide a more realistic view to make future business decision

B. Successes for the Client

Using our forecasts,

- The client brought down their resources by 26%
- Capacity planning and resource scheduling jobs for future dates are made easier
- Missing SLAs were brought down under 15%
- The client was able to lock resource hiring numbers with the vendors 2 months in advance.

VI. References

[1] https://www.statsmodels.org/stable/examples/notebooks/generated/statespace_sarimax_stata.html

[2] https://facebook.github.io/prophet/

[3] https://pypi.org/project/tbats/

Building Probabilistic and Isolated Learning models on Differentially private data for Campaign Optimisation in Programmatic setting

Manoj Kumar Rajendran
Principal Data Scientist
MiQ Digital India
manoj.kumar@miqdigital.com

***Abstract*— As we embark on an era of data-driven decisions, preserving the privacy of the underlying data is of utmost importance. Differential privacy has emerged as the go-to solution for tech giants and data vendors as it not only provides mathematical definition to privacy but also grants the ability to control the privacy parameter (noise) in the underlying data. The differentially private data aggregates sensitive individual data over multiple features and masks them with statistical noise.**

The conventional ML algorithms need to be tweaked to handle such aggregated data with noise. In this paper, probabilistic and Isolated learning techniques are leveraged to model differentially private data to improve the click-through rate of Ad-tech campaigns. The ability to predict the click event for different feature combinations such as URL, Ad type, Device type, etc. beforehand from the aggregated noisy data will help in developing a custom bidding strategy in a highly competitive programmatic setting (Real-time bidding)

Keywords—Differential Privacy, Aggregated Data, Real-time bidding, Probabilistic model, Isolated Learners

I. Introduction

The days of gut-based decision making are long gone, now we are in an age of data-driven, machine learning solutions. To let the algorithms arrive at such business-critical decisions, it is imperative to have a huge corpus of data and not miss even small variations from the expected trend. Various tech giants and data vendors are gathering users' personal and social interactions to accumulate a vast pool of usable data every second. Organizations are scrambling to make their datasets GDPR and CCPA compliant while struggling to protect the privacy of the opt-in customers when sharing and analyzing the data.

The Netflix 2006 challenge and NYC taxi data release are some of the instances in which the most common privacy preserving approach "Hashing/Anonymization" was exploited and brought to its knees. Through Reverse engineering and linkage attacks, the hackers showed the world that within a few hours it is possible to de-anonymize millions of customer data. Hence, we needed a modernized approach to cyber security that protects personal data far better than traditional methods. Differential privacy (DP) has emerged as a frontrunner in protecting sensitive data and provides strong mathematical guarantees.

Differential Privacy has been gaining a lot of traction in recent times and is the key focus area for companies in digital marketing and programmatic advertising which heavily relies on sensitive user behaviors/preferences in creating state of art predictive models.

As the world is making serious attempts to understand, implement and leverage "Differential privacy", some renowned institutions and tech goliaths have opted for this technique and have been reaping benefits.

- United states census 2020 data is "Differentially Private"
- Google has incorporated the "Local Differential Privacy" mechanism in building "RAPPOR" (randomized aggregatable privacy preserving ordinal responses) inside chrome to learn statistics on how unwanted software hijacks user's settings
- Apple has developed "Differential Privacy schema at scale" to study the user text and emoji patterns to improve the user experience

Most of the research on DP is done on the machine learning side and very little knowledge is shared on how to use the aggregated noisy DP data. In this paper, we will be discussing in detail two approaches to handle DP data effectively to solve one of the perennial burning problems in the Ad Tech industry – Improving the click-through rate of the Ads.

II. Differential Privacy

A. Concept and Definition

Differential privacy is a strong, mathematical definition of privacy in the context of statistical and machine learning analysis. This concept enables analysts to study information about the population of interest while masking the presence of an individual in that population.

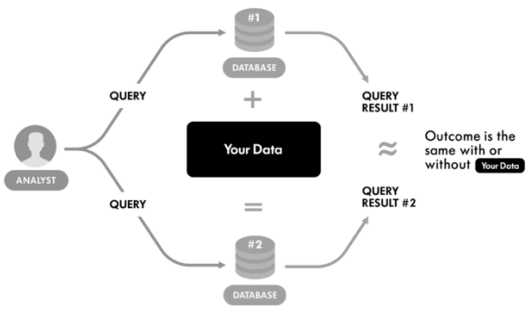


Fig. 1. Querying on Differentially Private Database

Imagine an analyst querying a database to count unique customers from a transaction table. Now, this query can be considered "Differentially Private" if by examining the output, an analyst cannot deduce whether an individual is present in the database or not. This intuitively means the presence of an individual shouldn't change the response/output of the query.

A process S is ε-differentially private if for all databases D_1 and D_2 which differ in only one individual for all possible outputs O:

$$P[S(D_1) = O] \leq e^{\varepsilon} \cdot P[S(D_2) = O]$$

ε is the privacy budget. Smaller ε makes $e^{\varepsilon} = 1$, hence the output probability distributions of function S in neighboring datasets D_1 and D_2 are roughly the same. Higher values of ε provide more privacy guarantees but affect the utility of the function (similar to bias vs variance). Depending on the trade-off between privacy and accuracy, differential privacy can be adopted globally.

B. Underlying Math

The crux of differential privacy is "adding carefully calibrated statistical noise" to obscure the underlying sensitive data. The two factors that decide the magnitude of noise are

- ε- The privacy budget (already discussed)
- Sensitivity of the query

Sensitivity is the extent to which the output of a query can change by the addition or deletion of a single individual from the dataset. For counting queries, the sensitivity is 1.

The noise that is added can be derived from Laplace, Gaussian, or Exponential distributions. Given a dataset D and the function $f: D \rightarrow R^d$, global sensitivity is Δf,

$$A(D) = f(D) + \text{noise}$$

Satisfies ε-differential privacy if the noise complies with the Laplace distribution with noise~Lap(Δf/ε), location parameter (LP) zero and scale parameter (SP) Δf/ε.

III. Handling Differentially Private Data

With many data vendors adopting DP in recent months, most of the sensitive data in the future will only be available in a differentially private manner (aggregated with noise). Hence the onus is on us to be prepared and research on how to use this data effectively. As this privacy-preserving framework provides the most sophisticated privacy guarantees and the noise is intentionally added to obscure sensitive data, the analyst has no say or information on the noise. If the privacy budget is made public, an analyst can have a rough expectation on the prediction models built on the DP data (obviously the performance of the models built on DP data will be lower than models built on clean sensitive data). Adopting differential privacy involves making predictions at the most granular level by modeling only the aggregated data.

The first solution that we present is a probabilistic approach – Naïve Bayes. The predefined packages in Python and R, build Naïve-Bayes classifier on granular data (at event level). The idea is to mimic similar steps but on the aggregated data. By estimating the event likelihood for all the features at different levels, it is possible to make predictions at the most granular level.

The second solution is inspired by the concept of "Federated Learning" (FL). In FL, individual algorithms are built on silos of data across multiple decentralized edge devices without exchanging data. Imagine this as a latitudinal learning process where data is cut horizontally into multiple independent blocks, meaning every model is built on all the features but with a different subset of data points. In isolated learning individual models are built on individual features separately, an equivalent of a longitudinal learning process with data cut vertically, meaning a model is built on one feature but with all data points. Researchers can argue that the interaction effects would be missed if features are modeled separately, so independent models are also built on the cross features (dependent variable aggregated by two features at a time)

The aforementioned solutions address only the aggregated nature of the DP data still, the "added statistical noise" part is to be tackled. As discussed previously, every time a DP result is presented to the analyst, random noise is added to the actual data (for the same query database returns different results), so it is impossible to replicate that noise in different data buckets to remove them. The idea of replicating and removing the noise refutes the concept of differential privacy. We have tried two approaches to correct the noise, which will be discussed in the subsequent section.

IV. Detailed Approach

Let $(x_i,y_i)_i \in_{1\ldots n}$ be a dataset D of n feature vectors $x_i \in X$ and binary labels $y_i \in \{0, 1\}$. As usual in supervised learning, we assume that examples (x_i,y_i) are independently sampled from an unknown distribution $\pi : X \times \{0,1\} \rightarrow [0, 1]$. We will also assume that examples are made of k categorical features. In our setup, the dataset $(x_i,y_i)_i \in_{1\ldots n}$ is not observed directly. Instead, we observe the count of examples and the sum of labels in several contingency tables.

select x_1, count(y_i), sum(y_i) from D group by x_1 with_differential_privacy (epsilon); (1)

select x_1, x_2 count(y_i), sum(y_i) from D group by x_1, x_2, with_differential_privacy (epsilon); (2)

TABLE I. Single and Cross Featuere Aggregations

	feature_1_value	feature_1_id	count	nb_clicks
1:	-96629	12	22.40	-5.66
2:	111058	12	92805.77	11223.71
3:	-86642	12	-4.91	8.74
4:	439517	12	4638.20	558.43
5:	-382286	12	-12.25	-4.76
6:	182391	12	37.83	5.06

	feature_1_value	feature_2_value	feature_1_id	feature_2_id	count	nb_clicks
1:	127803	-74587	8	16	-22.04	-16.36
2:	-245755	-84177	8	16	32.77	-6.83
3:	-473352	258050	8	16	674.52	151.97
4:	464036	408000	8	16	98.22	2.93
5:	-22347	17978	8	16	987.26	207.67
6:	359042	-389942	8	16	39.00	-4.23

Features are hashed, meaning an analyst cannot make sensible groupings, decisions based on the features. Negative values of counts are observed for some combinations as the added noise may be greater than the count values (noise is very significant in low count buckets and negligible in high count buckets). Certain features such as URL have very high cardinality (more than 1000 unique levels/values)

A. *Information value weighted Naïve Bayes Model*

Naïve Bayes is a conditional probability model, given a problem instance to be classified, which in our case is X, it calculates instance probabilities, $p(C_m| x_1,\ldots x_n)$ for each m possible outcomes or classes C_m. Using Bayes theorem, the conditional probability can be decomposed as

$$p(C_m|x) = (p(C_m)\, p(x|C_m))/p(x)$$

Assume that all features in x are mutually independent (which may or may not be the case in our click rate example) on the class C_m, then the conditional distribution is

$$p(C_m| x_1,\ldots x_n)= (1/Z)\, p(C_m)\, \pi\, p(x_i|C_m) \in_{1\ldots n}$$

where Z is a scaling factor and is a constant if the values of feature variables are known. The pseudo-code for the proposed approach is as follows.

1) Training data: X,y, contingency aggregation tables J_b
2) Calculate overall event rate and likelihood from the variable with the least cardinality
3) for each b $\in$ {1,2… k} do
a) Calculate likelihood w.r.t every single feature X_i
b) Estimate Information value (IV) for X_b w.r.t 'y'
c) If IV is significant, for every row in the scoring data X_{comb}, include that feature to participate in the calculation of final event and non-event likelihood, if IV is insignificant then exclude that variable in the calculation of final event and non-event likelihood

B. *Isolated learners*

Isolated learners in a nutshell mean "learning in isolation", one feature at a time. As mentioned earlier, both individual feature and cross-feature aggregations were used to arrive at the final prediction. Due to the high cardinal nature of certain variables such as URL, a sequential nearest neighbor approach was followed to reduce levels. This technique maps certain less frequent levels to dominant levels based on the target value (in our case click-through rate). The user is given a free hand to choose the maximum levels per feature to be allowed in the model building step. The pseudo-code for the proposed approach is as follows.

1) Training data: X,y, contingency aggregation tables J_b
2) for each b $\in$ {1,2… k} do
a) if X_b is highly cardinal, run a sequential nearest neighbor algorithm to reduce the cardinality
b) Build an Isolated GLM classifier and extract parameter coefficients for different levels of the feature X_b and store the model's log loss in a vector L (L_1… L_k)
3) For each row in scoring data X_{comb}, calculate prediction scores S_1…S_k using 'k' feature models and weight them with the inverse of their corresponding model log loss values stored in L to arrive at the final event prediction probability score S_{fin}

C. *Noise correction*

In Table I, we can see that the target variable "nb_clicks" representing total click count is negative. This happens due to the addition of random noise (negative value in this case) larger than the actual value itself (e.g., 4 - 20.36 = -16.36). It is imperative to account for the noise, if not, the dependent variable CTR calculated as nb_clicks/count will be negative. As different sets of random noises are added to every feature aggregation independently, the following techniques were tried on all the contingency aggregation tables separately.

1) Create biased samples from the published noise distribution (Laplace/Gaussian) such that all the negative data points turn positive, this approach is not favored for high cardinal features as a repeated sampling of noise is required to make all the data points positive
2) Find the largest negative value and reverse the sign and treat that as the white noise and add it across all the data points. Note that the white noise affects the data points with small values. High-value data points and low cardinal features are immune to this technique

V. Data and Experimental Setup

As of January 2021, there were 4.66 billion active internet users worldwide - 59.5 percent of the global population. Of this total, 92.6 percent (4.32 billion) accessed the internet via mobile devices. This huge digital traffic translates to a massive opportunity for the digital advertising domain. The world has made tremendous progress in identifying the customer needs/interests and delivering the right set of ads to enhance user experience and ROI for the advertisers. Click-through rate is a key metric in measuring the effectiveness of a digital Ad campaign. Predicting the probability of the click beforehand would save thousands of dollars for the advertisers. With millions of impressions data flowing in every second, it is impossible to study the pattern at the most granular level, hence there is a necessity to learn and predict at an aggregated level. With third-party cookies going away and privacy laws kicking in, a privacy component needs to be imbued in the process. This exactly fits in the "Differential Privacy learning framework" we are interested in.

To estimate the performance of our proposed solutions to improve the CTR of the campaigns, we pulled the count of impressions and clicks by single feature and cross features for the first two weeks of three different campaigns in the past. Some of the features including but are not limited to are URL, Ad size, Ad type, Browser, Ad position, Device type. Analyzing the data at the most granular level (by impressions) is a nightmare in terms of memory requirements and processing power, so it made absolute sense to pull the aggregated data with induced statistical noise of privacy budget ε =10 spread over 64 queries as in (1) and (2) – 8 single feature aggregates and 56 cross-feature aggregates (8*7). The aggregates take the form shown in Table I.

The exercise intends to identify certain feature combinations that are highly unlikely to result in a click and avoid bidding for them in the forthcoming weeks. We performed an EDA and identified that across campaigns only 4-8% of the unique feature combinations for which we bid, result in at least a single click. The input to the proposed models are clicks and no clicks count (no clicks = impressions – clicks) for different levels of individual feature and cross-feature combinations.

VI. Tests and Results

The proposed solutions were tested on three different past campaigns. Models built on features such as Ad position, Device type had significantly less log loss values (better

model fit) compared to the ones built on ISP and URL as seen in Table II. Even though we expect URL to be the critical factor in deciding the probability of a click, due to the high cardinal nature, the model built on URL was slightly underperforming. For simplicity, only the top single feature model performance is shown in Table II. In fact, the cross-feature models had better performance than some single feature models.

TABLE II. LOG LOSS VALUES OF FEATURE MODELS

Features	*Campaign 1*	*Campaign 2*	*Campaign 3*
Exchange Id	0.056	0.1086	0.133
Ad position	0.058	0.1089	0.137
Device type	0.069	0.1246	0.132
Browser	0.063	0.1322	0.146
Ad size	0.053	0.1385	0.132
Hour of the Ad	0.066	0.1476	0.142
ISP	0.078	0.1511	0.155
URL	0.096	0.1689	0.232

After building 64 isolated learner models (8 single and 56 cross-feature models) from the first two weeks of the campaign data, we used them to predict the probability of a click for all possible feature combinations in the third week.

Post scoring, we tagged the bottom 5-10% of the feature combinations as "Refrain" (avoid bidding) and the remaining combinations as "Bid" (continue bidding). As this is a backtesting exercise of past campaigns, we had third-week data without our proposed framework for performance studies.

Table III shows how the "Bid" bucket performed compared to the "Refrain" bucket in the third week of three different campaigns. For simplicity, we will be concentrating on the campaign 1 results as the same interpretation is applicable for the other two campaigns. If our proposed solution was in place at the end of the second week, we wouldn't have had this "Refrain" bucket. To assess how impactful the solution is, it is imperative to compare the "Bid" and "Refrain" buckets. On the upside, we would have avoided bidding for 7.7% of the total impressions (65,270) in turn saving 22.5% of the total Ad spending. On the downside, we would have missed out on the 58 clicks (3% of total clicks) but the stark difference in the CPC (cost per click) value indicates that we have spent $988 for every click in the "Refrain" bucket compared to $107 in the favorable "Bid" bucket. The cost of securing a click in the "Refrain" bucket is approximately nine times that of a "Bid" bucket. The CTR (Click through rate) and CPM (Cost per 1000 impressions) metrics are also worse in the "Refrain" bucket.

Based on the Advertiser's needs, the cut-off for defining the "Refrain" bucket can be changed. If the Advertiser is not keen on securing costly clicks, the cutoff can be pushed to 10-20% as in Campaign 1 resulting in huge savings. If clicks and impressions are important to the advertiser, the cutoff can be low (3-5%) as in Campaign 2 and 3. Note that in all the campaigns the proposed methodology has resulted in significant savings.

The solution is implemented using custom bidding scripts. The bid value for the "Refrain" combinations is made '0', meaning we will refrain from participating in the bids which we predict "to not result in a click". The Isolated learners approach outperformed the information value-weighted Naïve Bayes approach by a small margin. Hence only the results obtained from Isolated learners are documented here.

TABLE III. KPI COMPARISON OF BID VS REFRAIN FEATURE COMBINATION BUCKETS ACROSS CAMPAIGNS PREDICTED FOR THIRD WEEK

	Bucket	*Total clicks*	*Combinations*	*Ad Spend ($)*	*Impressions*	*CTR*	*CPC ($)*	*CPM ($)*
Campaign 1	**Bid (B)**	*1,851*	*17,860*	*197,489*	*778,852*	*0.24%*	*107*	*254*
	Refrain (R)	*58*	*2,035*	*57,287*	*65,270*	*0.09%*	*988*	*878*
	R/(B+R)	*3.0%*	*10.2%*	*22.5%*	*7.7%*			

	Bucket	*Total clicks*	*Combinations*	*Ad Spend ($)*	*Impressions*	*CTR*	*CPC ($)*	*CPM ($)*
Campaign 2	**Bid (B)**	*3,110*	*6,866*	*86,105*	*572,057*	*0.54%*	*28*	*151*
	Refrain (R)	*7*	*480*	*2,486*	*4,754*	*0.15%*	*355*	*523*
	R/(B+R)	*0.2%*	*6.5%*	*2.8%*	*0.8%*			

	Bucket	*Total clicks*	*Combinations*	*Ad Spend ($)*	*Impressions*	*CTR*	*CPC ($)*	*CPM ($)*
Campaign 3	**Bid (B)**	14,104	26,164	188,621	8,992,936	0.16%	13	21
	Refrain (R)	22	2,296	2,969	65,898	0.03%	135	45
	R/(B+R)	0.2%	8.1%	1.5%	0.7%			

VII. Conclusion and future work

In this paper, we have discussed how probabilistic and Isolated learning techniques can be deployed to efficiently model differentially private data and have demonstrated how they, in turn, improve the performance of the Ad campaigns. As we are shifting towards a privacy-first ecosystem, we must be prepared for a future with just aggregated data. Be it click prediction or bid allocation, the learnings from this work can be leveraged. With the cookie-less world edging closer to the programmatic advertising world faster than ever before, it is important to acquire knowledge to work on differentially private data and avoid being left behind. Given the niche nature of the problem we are trying to solve, the results are still satisfactory and provide us the scope to research and build models that can challenge and even surpass models built on sensitive granular data.

References

[1] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference. Springer, 265–284.

[2] Aggarwal, Charu C and Philip, S Yu. A general survey of privacy-preserving data mining models and algorithms. In Privacy-preserving data mining, pp. 11–52. Springer, 2008.

[3] McSherry F. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In: Proceedings of communications of the ACM, vol. 53(9), 2010.

[4] Kang M., Ramaswami G.K., Hodkiewicz M., Cripps E., Kim JM., Pecht M. (2016) A Sequential *k*-Nearest Neighbor Classification Approach for Data-Driven Fault Diagnosis Using Distance- and Density-Based Affinity Measures. In: Tan Y., Shi Y. (eds) Data Mining and Big Data. DMBD 2016.

[5] Codalab Criteo hackathon on handling differentially private data https://competitions.codalab.org/competitions/31485

[6] Blog by Manoj Kumar Rajendran (Author) on differential privacy https://medium.com/miq-tech-and-analytics/building-ensemble-of-iv-weighted-na%C3%AFve-bayes-model-on-differentially-private-data-to-predict-ad-841f918de79f

Models & Mechanisms for Motivating Machines

autonomous evolution of artificial intelligence

Prithwis Mukerjee
Praxis Business School, Kolkata
prithwis@praxis.ac.in

***Abstract*— We describe a mechanism that may allow digital machines to display intelligent behaviour without the overt intervention of human programmers. While most studies focus on how systems may be built to demonstrate intelligence, our focus is on how to motivate machines to demonstrate intelligence in new areas when they are not explicitly programmed to do so. Domain generalisation is an attempt in this direction but these attempts seek to enlarge capability in existing domains without moving into new domains of expertise. In this paper, we look at a set of existing software constructs, namely (i) the universally used TCP/IP protocol that connects machines, (ii) Docker containers that create portable programs, (iii) the Ramanujan machine that generates novel conjectures about number theory, (iv) blockchain technology that forms the basis of decentralised autonomous organisations (DAO) and (v) generative adversarial networks (GAN) that use pairs of generator-discriminator neural networks to create original content that is computationally indistinguishable from naturally occurring content. We show how these seemingly disparate dots can be connected to reveal a pattern that delineates the contours of a novel mechanism that may be used to define different levels of motivation for machines to demonstrate intelligent behaviour in new areas in an autonomous manner. We also speculate on how this may lead to a political colour in the behaviour that emerges through this process**

Keywords — machine learning, motivation, autonomous learning, Ramanujan machine, blockchain, GAN, dockers

I. Introduction

Artificial intelligence has made it possible for machines to display competence in many areas that humans and animals excel in, like recognising shapes, driving cars or competing with an adversary. However, none of these systems acquire these capabilities autonomously. They need to be 'motivated' to do so by being trained or programmed by humans.

Motivation is a mechanism that allows an artificial agent, which could be hardware or software, to behave in a manner that displays human traits like curiosity and the desire to explore. All such motivation mechanisms are eventually tied to some kind of a reward. In human beings, this reward is often ill-defined and non-material and could be rooted in inherent satisfaction like having fun or completing a challenge. This is intrinsic motivation as opposed to the motivation to perform under external pressure or direction.

The study of intrinsic motivation has its roots in psychology in the works of Berlyne [1] who explored ideas around familiarity and novelty. Festinger [2] viewed the process as being one that would reduce the dissonance between an external and internal view of the world as did Kagan [3] who sought to locate motivation in the desire to reduce the incongruence between experience and cognition.

Moving from human psychology to machine intelligence, Ouedeyer and Kaplan [4] characterised intrinsic motivation as being based on either knowledge, competence or morphology, where for all practical purposes, knowledge devolves into information and information theory. From an information perspective, the simplest possible approach is an objective function as is used in linear programming. However, for complex situations involving a large number of variables and system-states, a practical approach to defining and leveraging motivation is through reinforcement learning [5].

As a key tool in the toolbox of machine learning and artificial intelligence, reinforcement learning does away with the need for labelled or unlabelled data and depends instead on finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge). Because of its generality, it has been used in a wide range of disciplines that include game theory, simulation-based optimisation, multi-agent systems and swarm intelligence. In fact, Q-Learning and the concept of the Deep Q Network that use neural networks to improve the quality of action is perhaps the best way for a machine to learn to solve problems on its own without any human input.

However, one key drawback of reinforcement learning, in fact of all AI and ML techniques is the inability to generalize. These models may excel in one domain and the level of excellence can increase over time but like a fish out of water, these models are of little use in other non-related domains.

What we explore in this paper is the possibility of machines to explore, and acquire competence in, areas that lie beyond what they are currently competent in, and to do this without human intervention. We describe an evolutionary approach with analogies drawn from existing technologies like TCP/IP, Dockers, GPT3, the Ramanujan conjecture engine, Blockchain and a class of artificial neural networks called generative adversarial networks.

Software artifacts that display artificial intelligence are increasing in both number and sophistication. There are many definitions of what constitutes intelligence and there are many ways in which software has been programmed to demonstrate the same. Of all the many options, the use of artificial neural networks (ANN), that closely mimic the connectionist approach of animal brains, has been found to be most effective in performing tasks that are both useful and insightful. This includes, for example, recognising faces, driving cars, generating meaningful text passages and playing a wide range of games both against humans and against other programs. It may not be the case that the ANN

will always be the best way to demonstrate this kind of intelligent behaviour, so for the purpose of this study an ANN is neither necessary nor sufficient. All that we need is a digital artifact — a container of data, code, model, APIs or a combination of some of these — that we will refer to as a digital intelligence unit, or DIU. Having access to a DIU equips a digital computing device with the ability to demonstrate intelligence or mimic a specific behaviour of an intelligent biological system.

II. Digital Intelligence Unit

A DIU may be a digital construct but its input and output could be both digital as well as physical. A typical DIU that we deal with today may read in a piece of digital information, like an image or data file as an input and generate digital output, like a name or a class. However, there is no conceptual difficulty in assuming that the DIU is connected to sensors that capture a physical measurement from the environment or that it can cause wheels, drills, arms, actuators or tools to move and do physical work.

For example, a DIU may guide a car to move through traffic or terrain, pick up and assemble physical objects like rocks, machine components or even assemble two objects together. It can also sense and consume energy or where necessary cause the generation or transformation of energy from one state to another. Not all DIUs need to be very sophisticated. There could be very basic DIUs that simply interrogate other devices and exchange information or a DIU that allows a device to share a physical resource like a camera or a disk with another device. Nevertheless, we will treat the DIU as a digital abstraction that is resident on a digital hardware device like a computer.

A set of DIUs that work together may be viewed together as a larger, bigger or more sophisticated DIU. This larger DIU may still reside on one hardware device or its parts may be distributed across multiple hardware devices and identified with something like a Uniform Resource Identifier as is done in web development. However, this set of smaller, compatible DIUs is, conceptually, still another DIU. For example, a *rover* that NASA sends to Mars may consist of a collection of DIUs, each with its own intelligent function but the *rover* itself may be viewed as another DIU.

Continuing with this analogy, we may be tempted to view a biological animal, like a fish or man, as a DIU that is a collection of simpler DIUs. For the sake of argument, and simplicity, we may view, or model, a biological dog as a collection of four DIUs that can scan the environment and identify objects, distinguish between edible and non-edible objects, consume edible objects and generate signals that express facts about the taste of the food. So, four discrete DIUs are collected together to give a bigger DIU called a digital dog.

While this analogy appears tempting, it poses a few challenges.

III. Motivation

The first challenge is motivation. What motivates a DIU to demonstrate its intelligence? Or what is even more fundamental, what causes a DIU to come into existence?

For a DIU to demonstrate its intelligence, a program must be executed, which means that someone or something must start the program. This is not difficult because most digital platforms (as in computers with operating systems) have mechanisms that could cause certain programs (including DIUs) to start automatically when the system boots and then wait for signals, or interrupts, from the external world. These signals or interrupts could be key-presses or other events like the arrival of mail or the rise of temperature.

What is much more difficult is the process of creating the DIU in the first place. At our current level of understanding, the motivation to create a new DIU, or a new capability, lies in the hands of a human programmer and not within the domain of the digital device. As a human programmer, I can decide that in addition to recognising faces, we need to generate music for which we need an additional DIU. The process of building, or writing the code, for another DIU can be automated — programs to write other programs are not impossible with current technology, for example, the Github CoPilot or GPT3 — but someone must have the motivation to do so.

Current DIUs can be programmed to improve their performance with time. Face recognition programs or self-driving cars can be programmed to become better with use but we still do not have any logical mechanism through which a face recognition program suddenly decides to learn how to drive a car or vice versa. Coming back to our digital dog, its food recognition DIU can become better and better to differentiate good food from bad but it is extremely unlikely that it will acquire an additional DIU that makes it jump over the fence and search for better food outside the house.

Incidentally, a fence jumping DIU is not at all difficult to construct. With current technology it is a trivial exercise to build a robotic system that can jump over a fence. However, what is missing is the motivation to include this DIU in the current digital dog DIU and enhance its capabilities. We need a human programmer to identify this new need and add this additional DIU to the digital dog. Thus, the challenge lies in creating a mechanism, a motivation, that will allow the digital dog DIU to do this on its own.

Let us see how a biological dog does this in the physical world.

IV. The Community

A biological dog will jump the fence when it sees another biological dog jumping the fence. This ability to jump the fence, or the motivation to do so, is a behaviour, an ability (or intelligence unit), that resides within the community and which is acquired by or triggered in a member by observing other members. Perhaps this is far more so in humans than

in animals who are primarily hardwired. So the existence of a community is an important mechanism that allows an individual member to enlarge its DIU by acquiring the DIU available with some other member. In fact, in the story of human evolution, one of the reasons why humans have been so successful compared to other animals, is because they could form communities, share experiences and learn from each other. Henrich [6]

The first challenge is to devise a mechanism that motivates an individual member of the community to search for and get access to a DIU that is available with others. We refer to this as low level or primary motivation.

Computers that are connected in a network, say a TCP/IP based Ethernet network, can be viewed as a community that can share information with each other. However, they do not share information spontaneously. There needs to be a trigger activated by a human or an external stimulus that causes an exchange of information. Two computers, A and B may be connected by a network and one, say A, may have a DIU to recognise faces and the other, B, may have a DIU to drive cars. But there is no reason or likelihood for A to access the car-driving DIU on B or for B to access the face recognition DIU on A.

First, A would not be aware of the existence of the car-driving DIU on B and even if it were, there would be no reason, or motivation, to access it. To overcome this drawback and to create the mechanism for the primary motivation, we introduce the analogy of a computer virus.

V. Primary Motivation : The Virus and the DXP

A computer virus is a computer program, but we can view it as another DIU that is capable of performing at least two tasks. First, unlike other programs that sit patiently on their host platform waiting for a signal to do something, a virus program actively seeks out other devices in the network, or the *community*, and actively looks for exploitable exposure points. These exposure points could be TCP/IP ports through which messages could be sent or files and folders that can be written on. Second, it usually makes a copy of itself and places the copy in the second device.

Unlike the DIUs that we are interested in, a computer virus has malicious intentions but the principle under which they operate can be easily adopted by the DIUs. This leads to the idea of a DIU Exchange Protocol (DXP) that is built into the operating system of all digital devices, somewhat similar to the ubiquitous TCP/IP stack. The DXP stack on any host device, is designed to look into other, target devices connected on the network and if allowed to do so by the DXP stack on the target device, to scan it for the existence of new DIUs. This is the primary motivation built into the protocol stack. If a new DIU is found it will be copied back from the target to the host. Obviously, the process works in a symmetrical, peer-to-peer manner. Any machine with DXP can be a host and can pick up DIUs from any other target machine that is running the DXP protocol.

How do we know and recognise a DIU from the many other files, programs, images lying in the target? At the simplest level, a DIU can be identified by something like a file extension. For example, the *http* protocol recognises files with extensions like *htm*, *html* etc but will not interact with *doc* or *ppt* files. But given the complexity of a DIU, a simple file may not be sufficient. So, we may create a DIU as a container — a Docker container is a good analogy — that contains code with APIs, models and perhaps data that may be exchanged across devices. Markers present in the container, for example a correctly formatted XML file, will allow the DXP protocol to recognise them as such and distinct from other artifacts lying in the machine.

The process can be made significantly simpler and more secure if instead of exchanging DIUs from each other in a peer-to-peer manner, DXP protocol on each device publishes its DIU, or saves it, to a central location like a DIU repository. This would be analogous to the Docker hub, CRAN the repository for R packages or even GitHub which has a lot of source code. A better mechanism could be to store the DIUs as smart contracts in an Ethereum, or similar, blockchain. There are two advantages for using a blockchain based approach. First, blockchain *full* clients, that validate transactions and add blocks to the blockchain are designed to operate autonomously without any human intervention and as we show later, this is important in our scheme. Secondly, the DXP protocol that controls the process of adding a new block, with smart contracts, can be configured to include a validation process to ensure that only valid DIUs are added. The validation process will be explored in more detail later.

Once the repository is in place, then any member-device of the DIU user community can pull any DIU that is required or is of interest. The newly pulled DIU can then be assembled with other DIUs already present to create larger and more sophisticated DIU. This would not only mean that the device has evolved by acquiring a new ability but it has done so on the basis of its own primary motivation and not on the commands given by a human operator.

The DIU repository, and the primary motivation built into the DXP protocol, gives us a possible solution to the problem of how our digital dog enhances its ability by acquiring the ability, or DIU, to jump over the fence and find better food. This leads us to two more difficult questions, both of which are tied to the phenomenon of motivation. First, if it is not a human being, then who will create these DIUs and why? Second, why should an existing digital platform that already has a set of DIUs pull one more and add it to the DIUs that it already has.

We shall park the first question for the time being and focus on the second. Which DIU should a platform pull and why? What is the motivation for a device to pull a specific DIU? The basic or primary motivation, namely to scan the DIU hub and pull DIUs at periodic intervals, is baked into the design of the DXP protocol. But the choice of which DIU to pull depends on two factors, namely compatibility and utility.

VI. DIU Compatibility

For a DIU to work, it needs certain prerequisites. A DIU to drive a car needs access to a car, that is a device with engine, wheels, radars and many other things. A dog does not have wings and cannot fly in the air but it has legs that allow it to jump. So, it learns how to jump and not how to fly. Similarly, every digital device cannot pull any DIU. Its choice is restricted to a set of DIUs that it is in a position to operate, or for which it already has the prerequisite DIUs.

Prerequisites are usually chained backward. Let us consider that a device attempts to install a face-recognition DIU.

A face recognition DIU needs

- A DIU that already has the ability to access a network camera
  OR
- A DIU that can obtain a camera for the device, that in turn needs
  - A DIU that can execute an eCommerce transaction to purchase a camera and that in turn needs
    - A DIU that can earn money with say crypto mining or performing Amazon Mechanical Turk type assignments

    AND
  - A DIU that can physically plug a camera, that in turn needs
    - DIU that can operate a robotic arm, etc., that in turn needs
      - {another hierarchy of DIUs}

What if every device were to adopt this chain strategy? That would lead to a situation where every device can do everything which may not be physically possible or even desirable. Can a dog acquire the ability to fly? It may be possible after many generations -- as the evolution of species has shown -- but obviously the physical dog body will die but its genomes will get progressively altered over generations until it can fly. Similarly, the physical platform on which the digital device works may collapse but the software can get transferred from device to device and keep acquiring DIUs until it can do whatever it wants to do. This will take a long time and a lot of resources.

Instead, let us focus on how a device will pull a certain DIU that it wants to. But what is it that the device *wants-to-do*? This is a part of a larger question that will be addressed as the next level of motivation, or secondary motivation. Our current focus is on the question of *Which DIU should a platform pull?* and we said that the answer depends on compatibility and utility. We have addressed the issue of compatibility with primary motivation and we now look at utility and the secondary motivation.

VII. Secondary Motivation : Diu Utility

A DIU will be selected for a pull and implementation, if it provides some value to the device. A biological dog will learn how to jump because it gives it better food and so improves its ability to survive. It will not try to learn how to walk on two legs even if it sees another dog walking on two legs because walking on two legs does not increase its survivability. In the case of biological species, the utility of a particular ability is related to survival and this survival operates at different levels - survival of the individual body, survival of the species or the genome. There is also the possibility or the question of the survival of specific genes in the genome, if we agree to accept Dawkins' principle of The Selfish Gene.

Mapping this issue of biological survival to the world of digital devices is the next challenge and, in a sense, it loops back to the first issue that we identified already, namely motivation. We have already addressed this at one level of primary motivation that partially explains which DIU is to be pulled based on the ability to search for and pull DIUs on the basis of feasibility and compatibility. Now we need a next, or higher level of secondary motivation. Why should a digital platform seek any specific DIU to enlarge its set of DIUs?

In the biological world the only motivation behind the process of acquiring intelligence (or ability to perform certain tasks) is survival. Humans in a certain limited way are governed by Maslow's hierarchy of needs. When it comes to digital devices, we need to determine whether they should be guided, like lower animals, by the need to survive? Or should they be guided by something similar to the human hierarchy of needs? We know that in the case of computer viruses, the motivation is simply to spread to other machines, which is like a survival strategy. For a higher-level digital device, that is one with a complex DIU, the motivation could be something else.

So instead of trying to discover what could be the motivation, we can as humans build our own definition of secondary motivation directly into the algorithm of the DPX protocol. Note that humans are NOT building the DIU, but are designing the DPX protocol. Most optimization problems begin with a motivation, which is generally captured by means of an objective function that we try to minimise or maximise depending on the problem, but there could be others. The Open Shortest Path First is an algorithm that is baked into the heart of the Internet Protocol (IP) and determines the route to be taken by a data packet. Public Key Cryptography is an algorithm that is present in the *https* protocol and ensures data security. Proof of Work is an algorithm that is built into many cryptocurrency protocols to determine which block will be allowed to enter the blockchain.

Similarly, we need a motivation algorithm, the secondary motivation, that is baked into the DPX protocol that determines which DIU is of interest to the device or is useful. The design of this algorithm could be based on certain principles that human society holds dear, like the three principles of Utilitarianism that can be summed up as the *greatest good for the greatest number*. We could also draw upon certain ideas drawn from popular culture like the Three Laws of Robotics created by Isaac Asimov.

Obviously other competing approaches can be explored as well. All that we are saying now is that a motivation function, whatever it may be, needs to be built into the DPX algorithm and this will guide the choice of DIUs that will be allowed to be added to the repository or pulled from it by individual platforms.

With the algorithm of the secondary motivation that decides on which DIU to acquire, in place, we now have another question that we had parked earlier. If it is not a human being, then who will create this pool of DIUs and why? This leads us to a tertiary motivation that operates at the community level.

VIII. Tertiary Motivation : Community participation

Mutations that drive biological evolution occur at random. The ones that survive and are passed down through generations survive purely because they make the individuals *fitter* in their respective environments. Thus, evolution works in a brute-force manner, randomly trying out different permutations, and keeping only those mutations that survive the test of natural selection. Similarly, it can be possible to devise mechanisms that will generate newer and newer DIUs and then test them against the principles of secondary motivation. Here we will draw upon three analogies from the world of mathematics and computers and use them to define another level of motivation that can motivate the community as a whole to come up with more and more DIUs.

First let us consider the Ramanujan Machine that was created by Raayoni, et.al [7] to automatically generate new mathematical conjectures using an algorithmic approach. Ramanujan was an Indian mathematician who came up with many unproven conjectures, most of which were validated long after his death. However, these conjectures opened up new vistas in number theory that are still being exploited today. The Ramanujan machine is a network of computers running algorithms dedicated to finding conjectures about fundamental constants in the form of continued fractions. The purpose of the machine is to come up with conjectures (in the form of mathematical formulas) that humans can analyse, and hopefully prove to be true mathematically.[8].

The Ramanujan machine currently generates conjectures from a rather narrow domain of number theory and uses two algorithms, namely MITM and gradient descent. But we can envisage other algorithms that may generate tasks or objectives that are in line with the contours of the secondary motivation algorithm. Then the code for these tasks can be created by a product or process similar to the Open AI's GPT-3. This combination of a secondary motivation task generator and a code creator can then be viewed as a DIU engine that can run autonomously and generate any number of novel DIUs.

The second key piece of our strategy would be a blockchain based decentralised autonomous organisation (DAO). This is a self-sustaining distributed mechanism that creates economic value by encouraging individual machines to validate transactions — in this case DIUs created by the DIU machine — and rewards successful ones with cryptocurrency tokens. For a DIU to be valid it must meet the conditions of DPX protocol in terms of interoperability and the principles of secondary motivation. Only then it will be accepted as a part of the DIU blockchain and this blockchain will become the DIU hub or repository that we had discussed earlier.

Unlike the Bitcoin or current Ethereum blockchain that is based on an energy intensive Proof of Work protocol, this DIU Blockchain could be based on the principles of Proof of Stake or other energy efficient protocols.

This combination of a DIU generator and blockchain based DIU validator is remarkably similar in principle to the combination of generator-discriminator that is the basis of a class of artificial neural networks called generative adversarial network [GAN] first proposed by Goodfellow [9] A GAN, which is the third piece of our proposed tertiary motivation mechanism, is typically used to generate original artifacts that are nearly indistinguishable from similar artifacts that are found in natural populations. The most common example of this is human faces. Given a training set of human faces, a GAN can generate synthetic images of faces that are not found in the training set, but cannot be distinguished from naturally occurring images. In this case, the training set of DIUs could be the thousands of currently extant DIUs of AI systems that have been developed by humans. In fact, the blockchain could also be seeded by humans as in the first few thousand blocks could contain DIUs built from existing AI systems. However, going forward, the combination of the DIU engine and the blockchain validation process will create a GAN-like mechanism that will create a potentially endless series of DIU.

This mechanism will provide the tertiary motivation to fuel an evolving ecosystem of digital devices with more complex and useful DIUs. As a by-product, the crypto-tokens generated on this DIU Blockchain could be used by digital devices to pay for DIUs that they pull from the DIU hub.

IX. Conclusion

We have seen that human beings can build very useful software applications that mimic intelligent behaviour. We have also seen that it is possible for these applications to improve their performance based on observing the results of their actions. What has been lacking so far is an autonomous mechanism that has the motivation to create new software applications that mimic a range of intelligent behaviour that goes beyond what was originally coded for.

Biology has addressed this problem by creating new, random behaviours and then checking which of them are best suited for survival. This is called evolution and this biological evolution operates at three levels of survival - of the individual, of the species and of the gene.

We have proposed an analogous mechanism that can be used by digital devices to create a series of applications, the DIUs, that will automatically evolve in terms of complexity based on the three levels of motivation that we have described here.

Most of the software products that are used as analogies in this article already exist. What is missing is an algorithm for the secondary motivation that is stable but flexible and will work for all in a fair and equitable manner. In fact, the logic, or the algorithm that drives the secondary motivation on the DXP protocol is the key to the success of this scheme. This is where we leave technology and could wade into complex socio-political issues. Should this protocol have a socialist or a capitalist bias? This could be the basis of a fork in the DIU blockchain that leads to the emergence of two kinds of machines whose behaviour will reflect this underlying political bias. Even within the same blockchain, there could be DIUs that display either aggressive or conciliatory behaviour -- as is the case with self-driving cars. While the upload, or push, part of the protocol may allow both kinds of DIU to be accepted into the blockchain, the download, or pull part could have options that allow each device to pull DIUs that give it a specific *psychological* profile.

Acknowledgment

The first draft of this paper was reviewed by Subhayan Mukerjee, Assistant Professor, National University of Singapore and his valuable comments have been incorporated in the text.

References

[1] Berlyne, D.: Conflict, Arousal and Curiosity. McGraw-Hill, New York (1960)

[2] Festinger, L.: A theory of cognitive dissonance. Evanston, Row, Peterson (1957)

[3] Kagan, J.: Motives and development. Journal of Personality and Social Psychology 22, 51–66.

[4] Oudeyer, Pierre-Yves; Kaplan, Frederic (2008). "How can we define intrinsic motivation?". Proc. of the 8th Conf. on Epigenetic Robotics. 5. pp. 29–31.

[5] Kaelbling, Leslie P.; Littman, Michael L.; Moore, Andrew W. (1996). "Reinforcement Learning: A Survey". Journal of Artificial Intelligence Research. 4: 237–285. arXiv:cs/9605103. doi:10.1613/jair.301. S2CID 1708582. Archived from the original on 2001-11-20..

[6] Henrich, J : The Secret of Our Success. Princeton University Press, (2017) https://press.princeton.edu/books/paperback/9780691178431/the-secret-of-our-success.

[7] Raayoni, G., Gottlieb, S., Manor, Y. et al. Generating conjectures on fundamental constants with the Ramanujan Machine. Nature 590, 67–73 (2021). https://doi.org/10.1038/s41586-021-03229-4.

[8] https://phys.org/news/2019-07-ramanujan-machine-automatically-conjectures-fundamental.html

[9] Goodfellow, Ian; Pouget-Abadie, Jean; Mirza, Mehdi; Xu, Bing; Warde-Farley, David; Ozair, Sherjil; Courville, Aaron; Bengio, Yoshua (2014). Generative Adversarial Nets (PDF). Proceedings of the International Conference on Neural Information Processing Systems (NIPS 2014). pp. 2672–2680..

Driving towards building high performance Analytics Team: Strategic View

Shalini
Senior Data Scientist
ABB Bangalore, India

Rupesh Khare
Global Head - Advanced Analytics and AI
ABB Bangalore, India

***Abstract*— Engaging with multiple organizations trying to recognize and use predictive power of Analytics, compel us to bring all those journeys in a crisp way here. While we understand many organizations getting their hands dirty in analytics, but around 20% of them only succeed to relish real power of it.**

Some organizations start analytics with Intellectual curiosity, while some do it with few pilot projects, and few by targeting huge challenges in mind at start of journey. Our observations looking at diversified routes taken by organizations say, all they lack in common is structured approach and clear strategy. Organizations usually see their effort falling short in leveraging data available to them and becomes frustrated.

With this study, we propose generic analytics framework that can assist Leaders trying to establish new analytics team or reorganizing existing team. The first half of the paper revolves around framework that is result of extensive work experiences of the authors in starting a new analytics practice and observing counterparts in doing so. Typically, we understand C-suite recognizes great vision and key dimension, but nonetheless sometimes tend to ignore the structured framework of these dimensions and their execution. Our framework is here to help you ease out in identifying those dimensions and giving them a structure that could be apt for a user to apply in current situation. The later half of paper covers framework that is apt for an existing analytics team facing various execution challenges or struggling to grow up. Our work will navigate and assess the present scenario of team in terms of data size and Insights, and binding it with overall vision, it will recommend the route. The model and outcome from this model might look simplistic, but the impact it will have of the strategic vision and structured approach will be monumental.

The frameworks proposed can help company become Artificial Intelligence driven and hence improving performance

Keywords*— *Data Science, Artificial Intelligence, Analytics Framework

I. Introduction

We have been witnessing an increasing uptrend in several organizations exploring analytics, digging into predictive power, and striving to capitalize on it. The success of building an Analytics-Driven organization relies on many factors including having a high-performance team empowering this initiative. Many times, organizations getting their hands dirty into analytics, either end up with some pilot projects or fail to scale up to a significant level with minimal business impact noted. Quite often, many organizations getting their hands dirty into Analytics, either end up with some pilot projects or fail to scale up to a significant level. From a small-scale organization to a large-scale organization that engages in analytics set up, a plethora of them struggles to set up the right dynamics between brand, quality of delivery, and customers experience. And no wonder, trying to bring harmonization among these dimensions is fraught with challenges.

In this context, practitioners in Analytics and Artificial Intelligence domain often share their experiences in starting a new team or scaling the existing one up. Their observations cover success stories and often the ones which have not been so successful. These diversified experiences are quite enriching and powerful however they reach us a bit discreetly, and hence it is difficult to leverage them in practice. The situation identifies the need for a structured approach that can enable the leaders to strategically build an analytics team.

In this study, we propose a generic framework that would guide the leaders to strategies their plan to build a new analytics team. The framework is a result of the extensive work experiences of the authors in starting a new analytics practice. Typically, as we understand that leaders recognize the key dimensions of the strategy however sometimes tend to ignore the structured framework of these dimensions and their execution. The proposed framework identifies these dimensions and places them in a structure so that a user could be able to apply them to their existing work environment.

In the latter half of the paper, we take a view on an analytics team already in practice but facing challenges either in scaling it up or operating efficiently. This work first investigates and conducts an ‘As is” assessment of the analytics team and then based on the overall vision of the team, suggests a route to scale up or become operationally efficient. This model is simple to understand and to identify the present status of the team. Based on the outcome of the assessment, the leaders can sketch the strategy, in many cases using the framework illustrated in the following up section of the paper.

II. Scene Setting

Information Technology and data are widely available to all organizations. Even with that, managing humungous data is a persistent challenge, industry leaders are facing. Being on the journey of finding a better and optimized way of handling the massive data, has given birth to many buzzwords in the market “analytics”, “data science”,” Machine Learning”.

With so many software and hardware tools available in the market, market leaders must take a tough call based on various challenges and barriers available. Leaders are considered a connoisseur of this field, and hence are expected to use their experiences, business as well as technical skills to set up a high-performing analytics team, that can bring a revolutionary solution to the table; an analytics team, which could be characterized as visionary, pragmatic, and efficient enough from all angles of analytics solutions.

With this, comes multiple questions that are ought to be addressed for setting up a truly high performing and innovative analytics team:

1. Does the leadership and management levels of the organization ready to embrace change?
2. Do we have enough investment, direction, and authority from leadership?
3. Have we analyzed the analytics potential in various business areas and functions in the organization?
4. Have we accessed the low-hanging fruits and quick wins?

Fleshing out thoughts on above mentioned strategic points will help an organization on a smooth transition from inabilities to abilities.

III. Framework for a New Analytics Setup

To swim in the ocean of data, an organization that is planning to start a new analytics setup must carefully evaluate various dimensions of analytics strategy. Too often organizations build the setup based on a narrow and inflexible analytics strategy, which may not be efficient in an era of constant change.

It all starts when a business leader feels the need for analytics in his business. If the business seems to have many areas of improvement, people seem to be juggling in many business ways, which is not answerable through the current process. Furthermore, if one also wants to analyze business performance; access the business KPI's. tracking and forecasting the business KPI's. there, arises the dire and genuine need for Analytics.

Now when Analytics is figured out as a solution for all similar kinds of situations mentioned above, it's pertinent to outline broader strategic planning to bring the analytics team into existence. Moreover, it's apt to also prepare a laundry list of projects with all priorities defined into it.
What next? What should be the kick start point to address all? Speaking simple and straight, there is no standard way to answer this.
But as this paper is the outcome of our long-term experience and observations. Henceforth, the process portrayed here in form of a framework is comprehensive, capable, and practical enough to bring the best onto analytics at various levels.

A. Vision & Strategic positioning

"Vision plays a key role in producing useful change by helping to direct, align and inspire actions on the part of large numbers of people. Without an appropriate vision, a transformation effort can easily dissolve into a list of confusing, incompatible, and time-consuming projects that go in the wrong direction or nowhere at all"

The vision plays an indispensable role in bringing growth and change, by helping to direct and align to the people. On top of that, to strengthen feet in any changing environment, it is required to inspire, motivate, and connect to people frequently.

Here comes the first and the most crucial step of the framework where organizations address the questions like "analytics for whom?" and "what analytics product/services?". Some of these questions may look like this.

1. Who are the customers?
2. What are their business issues?
3. What business value do they expect to derive from your analytics function?
4. How closely linked is this to the corporate strategy?
5. How much demand is there for advanced analytics within/outside the organization?
6. What is the motivation and ability to sustain an advanced analytics function?
7. What kind of solution are customers looking for?
8. How complex and diverse is the analytics they are planning?
9. How complex is the deployment of the results?

The answers to these questions, not limited, will assess the analytics maturity and bind it with the overall direction and vision of the company.

Leaders and managers mutually have to anticipate the business problems and business area's pain points that can be targeted, as part of short and long-term goals. Anticipation of business problems helps in the identification of the right stakeholders. Furthermore, the right set of conversations with stakeholders can help in building up domain knowledge, which ultimately is a boon to your analytics journey.
In nutshell, understanding "what to do" and "for whom to do" is the key to bringing superior performance and increased profitability.

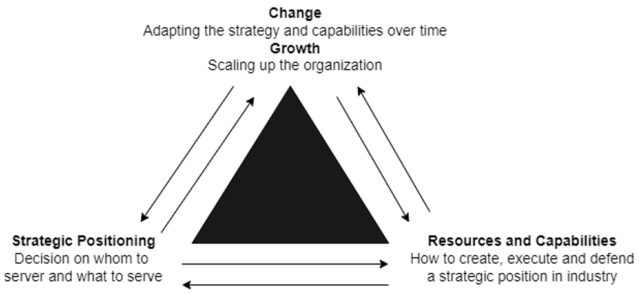


Fig 1. Framework for new Analytics Setup

B. Resource and Capabities

After developing the clarity on the questions in the previous segment of the framework, the managers get the base for the next step of strategy in which they find the answers to the "How to do" part of the framework. Here, the focus shifts to the Resources and Capabilities required to plan around.

1. Data Requirement
2. Data Depository
3. IT Infrastructure

4. Skillset and Culture
5. Data Security and confidentiality
6. Operating Model and Governance etc.

The data analytics team is more likely to succeed if the organization creates a "data foundation". Ensuring high-quality data should be a cornerstone of any data foundation. A strong data analytics infrastructure strategy ensures enhanced efficiency and productivity, makes collaboration much easier, and it allows one to easily access the information from anywhere provided one has the proper authentication steps in place. Proper data management system provides benefits such as operational cost reduction, a focused approach, higher efficiency, and flexibility to respond to changing business environments.

Further, it's very important to include not only people with Data Science or Data Engineering skills but also those with business and relationship skills. Data and analytics are most effective when world-class technology skills are paired with strong functional domain knowledge e.g. a Libbat. Believe it or not, this is the single most important person on the analytics team. A Libbat is a link between business and technology and the person who will define and deliver the most important analytical projects. The Libbat will combine critical key skills such as communications, business acumen, knowledge of the analytical process, and project management.

Organizations are investing in flexible data analytics operating models—featuring analytics-as-a-service—to glean customer and operational insights from their data. The data analytics operating model should be coupled with a governance structure that spans business and IT. Furthermore, it is focused on centralizing strategy, governance, and technology, optimizing the use of analytics talent, alleviating cloud data and data-in-motion security concerns, and monitoring data proliferation caused by businesses spinning up new cloud-based analytical tools and processes.

Identify key players and their key strengths, relevant to the team. More precisely, resource hiring either in-house or outsourcing, is going to give a real face to the analytics team. Gradually when the team grows, identification of skill gap and filling it up with continuous hiring will keep the team on top-notch. Given that technology is trendy and new, getting the best talent from the market becomes pivotal.

C. Growth and Change Management

This part of the framework deals with questions based on "where are we going" i.e. future direction and corresponding change management. The leaders evaluate their plans and actions considering the overall future business plans of the organization and focus on aspects such as.

1. Targeted growth in coming years
2. Any planned business expansion
3. Existing processes and culture
4. Leadership support to analytics
5. Future upskilling and supporting infrastructure

Successful change management methods involve preventing barriers to change while implementing change speedily and efficiently. Roadblocks to change could be anything from miscommunications to insufficient resources, but often the biggest one is cultural acceptance. For widespread adoption of change and a greater chance of realizing the ROI of analytic insights, it is important to focus on a few key change management principles such as inclusivity, transparency, investment in change management, flexibility, and continuous monitoring.

The key success factor in driving data-driven decision-making is the change in culture to embrace the need for analytics and artificial intelligence. The Data-Driven approach has to be part of the organization's strategy. It implies a shift from a traditional organizational culture to a data-driven culture, i.e., decision-making is determined by data rather than intuition, experience, or competitors' actions. It is popularly said, "Intuition may fail, but never the data". Data strategy is a cultural change that requires transformational leadership. The biggest obstacles to company transformation have a cultural component. The adoption of change by the people requires specific management that goes beyond words, technology, and talent itself.

IV. Framework to reorganize an existing Analytics Setup

Experiments and studies conducted by Gartner revealed that nearly 85% of the data analytics projects fail. Given its a shockingly large percentage, so the natural question that arises is why do these analytics projects fail and how can you avoid the same fate? While some consider this to be the sheer outcome of the size and complexities related to the seam, but the real picture is entirely different. It is not the technology but the implementation of the technology that leads to failures. Various studies and research examine these common pitfalls and how to maximize the chances of getting useful analytics for the company.

1. An Improper Starting Point
2. Lack of a Clear ROI
3. Forgetting to Involve Teams from All Areas of the Company
4. Poor Understanding of Data and its Sources
5. Trying to "Boil the Ocean"

This list may not be exhaustive, on the same note, there could be potentially more reasons affecting the success of analytics projects. Analytics projects need a clear focus and a logical starting point. It is important to include data from the right timeframe and the right sources; otherwise, it might arrive at an inaccurate conclusion. It's also vital that the team takes the time to establish a clear link between the analytics project and the bottom line. How is the central question or problem related to ROI? The link must be clear, strong, and well-

articulated. In most cases, an analytics project will have an effect on multiple divisions and teams within the company. So to maximize your chances of success, involve at least one representative from each affected team at the start of the2. project. A typical data analytics project requires you to pull data from many different sources. However, it's not uncommon for a large company to store data in multiple stand-alone reservoirs. This can make it challenging to access the data you need to succeed with your analysis. Also, the data analytics project needs to have a reasonable, achievable goal or objective. We can't do it all in one fell swoop. Many analytics projects are multi-phased, with each phase focusing on a specific aspect of a problem or issue. Once all phases are complete, the organization will have the insights the team needs to perform a more comprehensive analysis.

No wonder, the establishment of the right Analytics has mammoth potential, to generate business value and lasting effect in an organization's culture.

Our observation says organizations follow various roadmaps to accomplish it. Thinking organization is big, some leaders get aspirational looking at the size of data and vouch for analytics. While some leaders don't feel to start analytics as data seems to be small to them, considering the organization is just a start-up. Bringing all these kinds of observations together, we came up with a framework that can do justice to all sizes of organizations that are planning to reorganize the existing analytics set up more effectively and hence stronger insights on the table.

The figure below is the visualization of the framework we propose. The framework is based on the relationship between data size with the corresponding level of Insights drawn from it. Here are four states of that model, that can help in polishing and structuring the analytic setup efficiently:

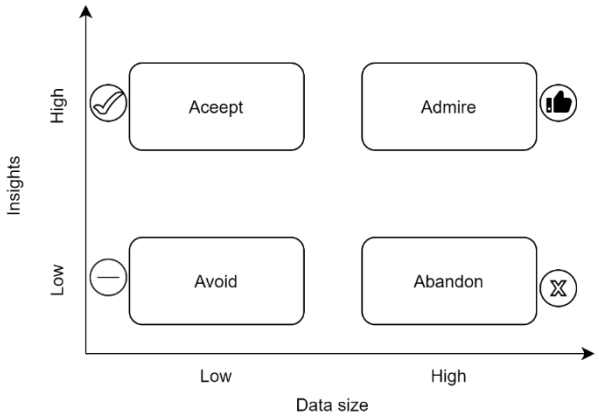


Fig 2. Analytics Success States

1. **Avoid (Low Data Size- Low Insight):** The organizations with small data sizes and not adequately capitalizing the data for the business insights fall in this category. Organizations should "Avoid" being in this situation, instead should start planning to apply analytics when the data size is small.

Accept (Low Data Size- High Insight): Organizations with proactive and visionary leadership start unleashing their data's full potential even when they don't hold a great amount of data. Gradually, they land into a situation, where they try to catch up with new things/technology. Now as they accept the potential of analytics, they go to acquire the best skills and make things operational.

3. Abandon (High Data Size- Low Insight): This is a typical state of many large/medium size but laggard organizations. Over the period their business grew and hence the data, however, they didn't really take initiative to use this data to their advantage. The data lied almost untouched or abandoned. With enormous data availability, sometimes organizations get aspirational and try to "boil the ocean" by taking some major steps to use data for bigger insights. This jerky move has little chance of success unless the building of a new analytics practice is methodically planned and executed.

4. Admire (High Data Size- High Insight): This is like a "Moksha" state for any organization. In this admiring state, they take full advantage of available data. It is observed that these organizations are proactive and flexible in embracing the new technology, process, and any other change which may impact their business positively. These organizations closely observe and look around the market, the innovative way it's transforming the industries, and hence quite often use analytics in this pursuit.

After having understood various states an organization may travel during its journey in analytics, it's now the time to investigate the options an organization may want to exercise when it realizes that its analytics setup is not delivering as planned. Most of the organizations face as discussed above, this situation and then try to do course correction. In order to facilitate this, we are trying to sketch below, using the frameworks presented in this paper, a broader guideline that may help these organizations to conduct a structured root cause analysis followed by a well-informed action plan.

1. In the long term, most organizations target "Admire" as their destination.

2. In case of re-organizing the analytics set up, assess the current state and spot its position in the framework illustrated in fig 2. Though an organization may occupy any of four states shown in figure 2 however, it is observed that if the existing setup fails, then it is highly likely that the setup is in either "Avoid" or "Abandon" state. It is less likely that a setup that comes in either the "Accept" or "Admire" state is failed.

3. Once the current state is established, try to understand the probable reason(s) of the failure. Analyze the reasons in the light of the framework presented in fig 1. The leaders may wish to get the answers around vision, culture, data, skillset, infrastructure, sponsorship, the scope of the work, customers, processes, etc.

4. After understanding the gaps, the leaders need to go to the drawing board again and start thinking fresh. The drivers of the failures may be many and different depending upon the context. For Instance, a setup in the "Avoid" state may realize that it didn't plan to leverage data until now and may want to take actions that will take it to the "Accept" stage. On the other hand, leaders of the setup in "Abandon" state may figure out they tried to 'Boil the Ocean". This may lead them to travel from "Abandon" to "Accept" by scoping the "manageable" targets and first set the tune of the success before flying high towards "Admire'. There could be many more such situations that demand a relook. In the journey of reflecting back on starting new, the framework in Fig 1 can be extremely useful.

V. Some DOs, DON'Ts and Myths

The primary mission of many data and analytics leaders and chief data officers (CDOs) consists of building an effective data and analytics team to establish data and analytics as a strategic discipline within their organization. However, as discussed earlier, this mission may not be accomplished if leaders commit some fundamental mistakes in the planning and execution of the strategy in building analytics setup. We are summarizing below a quick summary of some DOs and DONTs:

- Do evaluate the current process and identify areas of improvement but don't rely on one plan/strategy
- Do perform budgeting and cost analysis but don't come down on project execution level, till good confidence achieved on analytics team built
- Do strive for data-driven culture but don't overlook infrastructure
- Do prioritize projects and initiatives but don't ignore data maturity/maturity assessment within the organization
- Do be in continuous touch and discussion with leadership to be aligned with the organization's business and vision but don't hesitate in seeking guidance and help on accessing data, processes, and technology
- Do emphasize using basic in-house tools, open-source tools and slowly move on in including others paid/licensed tools gradually, as required but don't "Boil the ocean"

Let's take you through some common misconceptions about analytics setup.

MYTH 1:
One needs to be Ph.D. and rocket scientist to setup Analytics.

⇨ As far as one understands the right data, tools, technology, people, process, and business, one need not be concerned about the degree he possesses.
The reality is one needs to understand their customer and their expectations well. One should know the right set of data and processes to convert it into actionable insights.

MYTH 2:
One needs to have availability of all highly sophisticated tools available in the market to kick start the Analytics.

⇨ There is no hard and fast rule of starting any analytics team with such highly sophisticated and high-priced tools.
One can always start with open-source tools like R, Python, Julia, etc. Gradually when things start getting momentum, things can be looked around for more sophisticated tools aligning with the business need.

MYTH 3:
Only a big organization can setup analytics and reap its benefit.

⇨ All scale companies can setup analytics teams and benefit from that.
Once data sources are recognized, one can pull data and leverage analytics to come up with insights. Businesses should be flexible enough to try out various approaches to data until they land on satisfactory results.

MYTH 4:
Being an expert in Data Science and technical aspects can bring you quick success with analytics.

⇨ Setting up a global and highly efficient Analytics team demands excellence in technical as well as business.
Once we understand business, we are able to understand its problem area and pain points. Now when pain points are in hand, one can analyze various data science approach suitable in that aspect.
Henceforth, End to end understanding of the project requires both the feathers in your cap.

Conclusion

Armed with data, a structured approach, and a concrete strategic plan to set up an efficient analytic team can do wonders with business and organization. The framework,

model, and points discussed in this paper give a pragmatic view of all possible cases, and insights around that.

We discussed the analytics frameworks, which target to answer the point from all dimensions. This enables us to now assess the current state of data, size, and process of organization, providing further guidance to move towards the next level of maturity. We also understood various paths taken by the company and their upshots. Undoubtedly, Myth buster gives an added advantage on similar thoughts. All this concludes with efficient analytics and helps organizations run better.

REFERENCES

[1] https://online.hbs.edu/blog/post/analytics-team-structure

[2] https://mitsloan.mit.edu/ideas-made-to-matter/how-to-build-a-data-analytics-dream-team

[3] https://quanthub.com/advanced-analytics/

[4] https://www.mckinsey.com/industries/financial-services/our-insights/building-an-effective-analytics-organization

[5] https://www.cio.com/article/3639911/the-secrets-of-highly-successful-data-analytics-teams.html

[6] https://www.arkatechture.com/blog/data-analytics-infrastructure

[7] https://www2.deloitte.com/us/en/pages/consulting/articles/data-analytics-operating-model.html

[8] https://www.logic2020.com/insight/analytics-driven-change-management

[9] https://www.sdggroup.com/en/insights-room/data-driven-mindset-strategic-formula-cultural-transformation

[10] https://www.saviantconsulting.com/blog/why-data-analytics-projects-fail.aspx

[11] https://7t.co/blog/5-reasons-why-analytics-projects-fail/

ADDS - Attention-based Detection and Trajectory Prediction in Counter-Drone Systems

Swadesh Jana
Jadavpur University
Kolkata, India
swadeshjana@gmail.com

Sk Shahnawaz
Jadavpur University
Kolkata, India
skshahnawaz2909@gmail.com

***Abstract*—The usage of drones or Unmanned Aerial Vehicles (UAVs), both for military and civilian purposes, has increased in India in the past decade. They are being used for reconnaissance, imaging, damage assessment, payload delivery (lethal as well as utilitarian), and recently among the COVID-19 pandemic, for contact-less delivery of medicines. As UAVs are getting more affordable and easier to fly, and more adaptable for crime, terrorism, or military purposes, defence forces are getting increasingly challenged by the need to quickly detect and identify such aircraft. A number of counter-drone solutions are being developed, but the cost of drone detection ground systems can also be very high, depending on the number of sensors deployed and powerful fusion algorithms. With recent research showing attention-based neural networks possessing better object detection and tracking capabilities than traditional convolutional neural networks, a novel attention-based encoder-decoder transformer model is developed to detect drones in a given field of vision and track their future trajectory in real-time. This allows surveillance systems in red zones (such as military bases, airports, etc.) to detect and neutralize possible dangers associated with drones.**

Keywords—drones, machine learning, object detection, transformers, tracking, security

I. Introduction

Unmanned aerial vehicles (UAVs) or drones have become popular as an utility in recent times. While around the world, there has been an increased usage of drones, in India, the recent changes in government policies have facilitated the growth of the drone market to a multi-billion dollar in the upcoming years [1, 2]. Being a remotely controlled or automated aircraft, UAVs can be used in a plethora of activities as food delivery services, blood transport or even, performing rescue operations in remote and hostile areas. Drones fitted with cameras can be used in the police departments, geographic mapping and inspection of high towers and buildings [3]. With technological advancements, drones have gathered the ability to become faster and lighter along with being accurate enough to autonomously fly in confined spaces and in close proximity to people [4].

However, with such advances in this field, UAVs have also become a threat to the public and military security. In 2019 USA successfully conducted a test when the MQ-9 Reaper, a large UAV was able to shoot down another drone by firing a missile [3]. In 2021, the Air Force station at Jammu was involved in a terror attack, where two drones released explosives over the base mildly injuring two personnel. Additionally, it was reported that there have been 77 recorded sightings of drones in the Western Front in 2020 alone [1]. As explored in this article [3], drones with heavier payloads and surveillance sensors and cameras are being developed and deployed for military use. With China emerging as a leader in exporting low cost armed and stealth drones, it is increasingly becoming relevant to develop counter-drone systems for national protection and well-being.

With the emergence of artificial intelligence (AI), diverse activities are increasingly getting automated with precise and accurate AI solutions [5]. This motivates the use of AI and computer vision (CV) in high security tasks such as counter-drone systems. CV enables the detection and tracking of drones using images and videos. Any camera capable of taking high resolution images or videos can be utilised to train AI models for this purpose. Drones can be efficiently detected in the range of view of the camera and a series of video frames can be studied to predict the future movement direction of the detected drones. Self-attention-based architectures, in particular Transformers [6] have recently been adopted as a model of choice in various AI tasks such as convolutional neural networks and natural language processing. As shown by [7], vision transformers achieve good results on benchmark datasets such as ImageNet and CIFAR. Essentially, attention-based transformers are fed with linear embeddings developed from 2D patches of the images and then these embeddings are treated as tokens on which the model learns inductive biases. To solve long-term predictive reasoning tasks, standard architectures are used to extract frame or clip level information, which are then aggregated using clustering [8, 9], recurrence [10, 11], or attention-based models [9, 12, 13]. Except for recurrent models, most of these models simply collect features over time, with no attention for simulating the video's sequential temporal progression over frames. While recurrent models such as LSTMs have been studied for anticipation, their sequential structure means they struggle to describe long-range temporal relationships [14, 15].

Anticipative Video Transformer (AVT) is an alternative video modeling architecture that substitutes "aggregation" based temporal modeling with an anticipative architecture. To address the aforementioned tradeoffs, the proposed model leverages the sequential nature of videos while limiting the drawbacks associated with recurrent architectures. AVT, like recurrent models, may be used indefinitely to forecast further into the future (i.e. produce future predictions), but it does so while simultaneously analyzing the input with long-range attention, which is typically lost in recurrent architectures [9].

The paper is divided into the following sections. Related work in the area of counter-drone detection systems has been discussed in section II.

II. Related Work

Several methods exist in the space of drone detection system. The equipment used in the suggested approaches in the market and academic literature may be divided into five categories: RADAR, LIDAR, acoustics, RF signal detection, and optics. RADAR technology has been used to identify aerial vehicles for decades; however, traditional RADAR systems are not capable of detecting tiny commercial UAVs. Furthermore, they are flying at considerably lower speeds, which reduces the Doppler signature. Despite the fact that such examples exist [16]-[18], they often fail to categorize other airborne objects such as birds and background clutter owing to their enhanced sensitivity in this circumstance. On the other hand, LIDAR based systems [19] [20] are vulnerable to lighting conditions and producing voluminous data. RF signal analysis, which aims to capture the communication between the drone and the ground operator, is the most prevalent approach for drone detection. The biggest problem with this strategy is that the drone might be flown without any ground control and on a pre-programmed flight path. Acoustics has been used also to detect drones by employing microphone arrays [21] [22]. The goal is to classify specific drone rotor sounds, but they fall short of high precision and operational range. The maximum range of audio-assisted systems is between 200 and 250 metres. Another problem is that the technology is not feasible in metropolitan or noisy locations like airports.

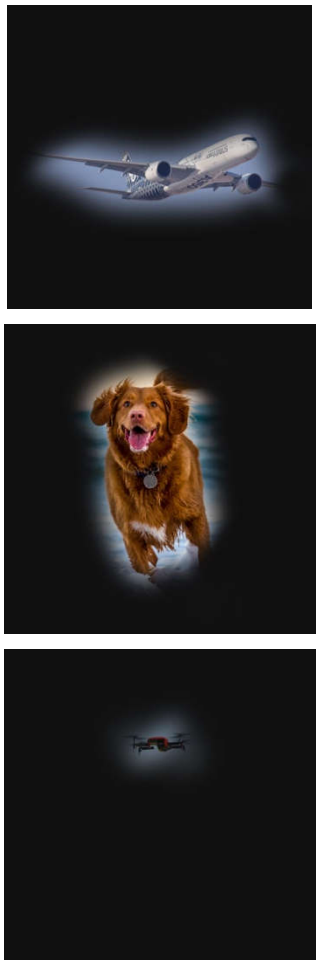


Fig. 1. Attention masks generated by transformer models.

Optics stands out among the other approaches to drone detection that have been discussed earlier. Because of its robustness, accuracy, range, and interpretability, optics has been viewed as the most practical solution to address this problem [23]. Deep learning for computer vision based on convolutional neural networks (CNNs) has already become the de facto approach for detection and recognition tasks, thanks to the convergence of open source data (i.e., photos, videos), developed algorithms, and affordable GPU resources [24] [25]. Deep learning is used in the majority of publications published in the last few years proposing to employ computer vision for autonomous drone surveillance tasks [26]-[28].

In drone tracking systems, historical data is required to predict future motion. In such a scenario, a temporal prediction system for action anticipation needs to be developed. Action anticipation is the task of predicting future actions given a video clip. For this job, a variety of methods have been proposed, including learning representations by predicting future features [29] [30], aggregating prior features [13], and leveraging affordances and hand motion [31]. Our research proposes a novel approach based on sequential regression that focuses on long-term prediction. Self-supervised video feature learning algorithms learn representations from unlabeled video, which are then fine-tuned for specific downstream tasks. While this aspect of ADDS shares motivation with prior and concurrent work [30] [32] [33], our architecture for achieving predictive features is distinct (transformer-based rather than convolutional/recurrent), it operates over raw frames or continuous video features rather than clustered 'visual words' [34], assumes only visual data, and is jointly trained for action anticipation. While self-attention techniques like transformers [35] have shown to be effective for high-level vision reasoning problems, transformers are being largely used to totally replace convolutional networks in image recognition [36] [37]. Prior work in the field of video has primarily relied on attention structures [30] [38] built on top of basic spatio-temporal convolutional base architectures [39]. ADDS, on the other hand, is a video end-to-end transformer architecture. Unlike concurrent approaches [40], which are bidirectional and focus on classical action detection, ADDS uses a causal framework to construct future trajectories for targeted UAVs.

Our work contributes a new video architecture for anticipation and predicting the future trajectories of motion of drones, and we demonstrate its promising advantages in designing Anti-Drone systems used in defense.

III. Proposed Model Architecture

We propose a multi-agent framework in which each object of interest (attention) is represented as an instance of our transformer network. As a result of its previous motion, the Transformer Network predicts the drone's future motion. The deep learning network used in this paper is divided into the *backbone*, which is fed with the image frames and the *head*, which gathers the encoded features and predicts the future motion. We describe in this section the encoder-decoder Transformer Network and the model input and output.

Fig. 2. Timeline of frames observed and predicted

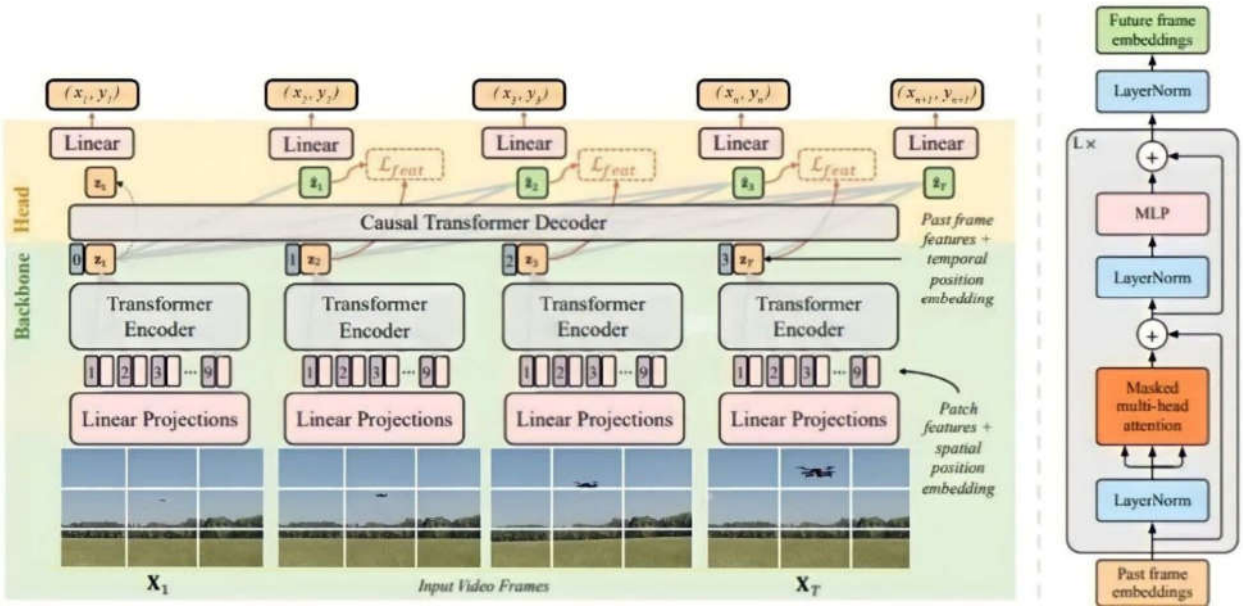


Fig. 3. **Model Architecture.** (a) The transformer encoder-decoder network that takes input video frames and gives output the predicted (x, y) drone coordinates. (b) Structure of the Casual Transformer-Decoder

A. Trajectory Prediction System

While multiple anticipation problem setups have been explored in the literature, in this work we follow the setup defined in recent challenge benchmarks [41] and illustrated in Figure. For each action segment labeled starting at time T_S, we aim to recognize it using a T_O length video segment prior to it, i.e., from T_S-T_O to T_S. Observed segments can be of any length, the future possible positions of the drones are predicted during the time T_a. For each of the subsequent frames, the transformer performs a regression task by predicting (x, y) which indicates the position of the drone in the next frame.

B. Backbone Network

A video clip consisting of T frames such that $V = \{X_1, X_2, X_3, \ldots, X_T\}$ forms the input to the backbone network, B. For each frame (ViT) [7] to extract feature representations z_t and $z_t = B(X_t)$. The transformer [6] is capable of extracting features from the image frames to perform as well as a convolutional neural network and also generate temporal embeddings that help to perform time series analysis of the data.

The transformer model that has been chosen from the ViT group of networks from this paper is the ViT-B/16 architecture. Each input frame is split into 16 X 16 non-overlapping patches. These patches are flattened into a 256D vector, which is then projected linearly into 768D feature dimensions to be input to the encoder network. Finally, the spatial position embeddings of the frames are added to the frame-specific position encodings, in order to apply the same backbone model to each frame. This is then passed through the transformer encoder [6] with pre-norm [35].

The transformer encoder is a stacking of alternating layers of multi-headed self-attention (MSA) [6] and multi-layer perceptrons (MLP) [42]. As used in [6], layernorm (LN) is applied to every block and residual connections after every block [35] [43]. The MLP layers are local and capable of baking in inductive biases similar to CNNs while the self-attention layers are global. The transformer encoder has shown image processing capabilities equivalent to that of CNNs [44] [45] and therefore are a choice of backbone network in our time series model. The output from the encoder model i.e. $\{z_1, z_2, \ldots, z_T\}$ form the input to the head network.

C. Head Network

A Casual Transformer Decoder, D [9] is utilised to predict the future features for each input frame. The temporal position embeddings are appended to the features extracted by the encoder and fed into one single decoder. The outputs are as follows:

$$\widehat{z_1}, \widehat{z_2}, \ldots, \widehat{z_T} = D(z_1, z_2, \ldots, z_T) \quad (1)$$

Here, the decoder acts on the frame features z_t as well as the features preceding that frame to get the output $\widehat{z_t}$ These outputs are then passed through linear feed forward networks to get the output $\widehat{y_t}$. This network performs like a regression model on the features extracted by the decoder to predict the coordinates of the drone in the next frame. In other words, $\widehat{y_t} = (x_i, y_i)$, where (x_i, y_i) are the coordinates of the drone with respect to the top left corner of the image. By utilizing successive predictions of the model, a sequence of regression tasks can be performed by appending the predicted features to previous frame outputs to obtain results further into the future.

The implementation of D is illustrated in Fig. 3. The modified embeddings (temporal positions + encoded features) are passed through several consecutive decoder layers, each consisting of masked multi-head attention, layernorm (LN) and a MLP and finally through a LN to obtain future frame embeddings.

IV. Experiments

Using the ADDS network described above, experiments are performed on a standard drone tracking dataset. In this section, the experiment details, environment, dataset, and training procedures have been described.

A. Dataset

The drone dataset used for our experiments has been compiled by [47]. Each of the collected videos consists of a camera fixed in position facing a drone that is flying in a region of about 100 X 100 m, at heights of up to 50m. The data is annotated to get the 2D locations of the drone with respect to the image boundary. A detailed study of the video capture mechanism has been done in [47]. For this work, a window of past frames is taken to predict the future positions. A sliding window mechanism on the whole video is used to get T frames and predict the position of the drone in T+1 frame. If there is no drone in any given frame, then the window is discarded. Further, pre-processing of the images is done by scaling them to obtain image frames with mean 0 and standard deviation 1.

B. Experimental Setup

The experiments have been performed on Jupyter notebooks provided by Google Colaboratory with NVIDIA Tesla P100-PCIE GPU runtime. The code has been written in Python 3.9 using deep learning libraries. The following additional modules have been used: Numpy (1.21), Scipy (1.7), Pillow (8.4.0), Matplotlib (3.4.0), scikit-image (0.19), opencv-image (4.5.2), and imgaug (0.2.9).

C. Training

The ADDS model developed for future tracking drone coordinates in the frame is trained using a clip of video frames containing a drone. In order to supervise the predictions made by the model, a L2 loss function is used.

$$L\big((x,y),(\hat{x},\hat{y})\big) = (y-\hat{y})^2 + (x-\hat{x})^2 \tag{2}$$

, where $(\hat{x},\hat{y})$ is the predicted coordinates of the drone. Additionally, the *feature* level of the model is also supervised to train the casual structure of the decoder.

$$L_{feat} = \sum_{t=1}^{T-1} |\hat{z_t} - z_{t+1}^2| \tag{3}$$

This has been taken from the paper [29], which have effectively shown that self-supervision can performed by anticipating future visual representations.

As shown previously [9] [46], transferring the weights from a previously trained model helps to boost training. Therefore, the model is initialized with pre-trained weights from the ImageNet dataset and then finetuned for our prediction task. The encoder is a 12-layer block of network that operates on the 768D image representations. The head used is a 6-layer model with random weights. The model is trained via backpropagation with the Adam optimizer. The learning rate is warmed-up for the first 5 epochs, kept at the maximum for 10 epochs and subsequently exponentially decayed.

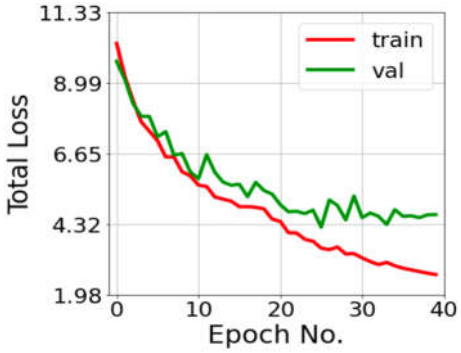


Fig. 4. Graph depicting the total loss

The loss generated per epoch while training the model have been depicted in Fig. 4. As seen after about 30 epochs, the validation loss starts increasing and overfitting is observed. Therefore, we stop training and test it on a new video. The route prediction is shown below in Fig. 5.

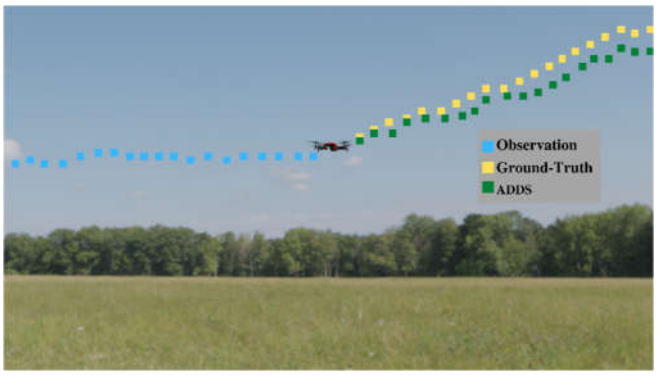


Fig. 5. Route Prediction performed by the ADDS transformer model

V. Anticipation has its limits

Although the ADDS transformer model has proved to be a reliable solution to produce better counter-drone systems, uncertainty in future trajectories such as the following can reduce its accuracy. As earlier stated, drones have some utility such as surveillance or defence systems and therefore, it is impossible for the ADDS system to identify the long term goals of the detected drones. Additionally, drones can be made to randomly follow certain sharp trajectories that is impossible to predict. Drones can also be hidden from view by structures or trees, which gives them the opportunity to intentionally follow such a hidden from view path. These possible loopholes in the system can be used by enemy drones to attack targets without getting detected. One possible solution is to additionally use a RF system that can work in tandem with the ADDS system.

VI. Conclusion and Future Work

The ADDS vision transformer network developed in this paper is a novel one for detection and tracking of drones in real-time in videos. It can not only perform as well as convolutional neural networks but also make use of the temporal relationships to predict future positions of the drone. In this paper, we extensively studied the structure of the

transformer model that can be used for our purpose and also, the tracking capabilities demonstrated by it.

As our future work, we would like to improve our ADDS system to be able to track multiple drones. Although, it has been previously shown capable of detecting multiple objects in images, using transformers as a tool for tracking multiple drones has not been conducted before. In a surveillance system, the added capability to detect and track multiple drones is of foremost requirement. As an additional capability, ADDS can be used to classify malicious drones from friendly ones and raise alarms. By further deploying the model to remote drones as well as ground-based surveillance systems for real-time trajectory prediction and tracking, ADDS can become a fully functional system that can be used to detect and track drones in an area. Surveillance systems of the government and military can be aided by incorporating our system.

REFERENCES

[1] P. Krishnankutty, "Why india's drone market could become a multi-billion-dollar industry in next decade," *The Print, India*, 2021, accessed on 2021-12-07. [Online]. Available: https://theprint.in/india/governance/why-indias-drone-marketcould-become-a-multi-billion-dollar-industry-in-next-decade/700817/

[2] M. J. Kumar, "The sky is not the limit: The new rules give wings to the drone technology in india," *IETE Technical Review*, vol. 38, no. 5, pp. 463–464, 2021. [Online]. Available: https://doi.org/10.1080/02564602.2021.1983967

[3] J. P. West and J. S. Bowman, "The domestic use of drones: An ethical analysis of surveillance issues," *Public Administration Review*, vol. 76, no. 4, pp. 649–659, 2016.

[4] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.

[5] G. Misuraca, C. van Noordt, and A. Boukli, "The use of ai in public services: results from a preliminary mapping across the eu," in *Proceedings of the 13th International Conference on Theory and Practice of Electronic Governance*, 2020, pp. 90–99.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[8] R. Girdhar, D. Ramanan, A. Gupta, J. Sivic, and B. Russell, "Actionvlad: Learning spatio-temporal aggregation for action classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 971–980.

[9] R. Girdhar and K. Grauman, "Anticipative video transformer," 2021.

[10] A. Furnari and G. M. Farinella, "What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6252–6261.

[11] A. Furnari and G. Farinella, "Rolling-unrolling lstms for action anticipation from first-person video," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[12] X. Long, C. Gan, G. De Melo, J. Wu, X. Liu, and S. Wen, "Attention clusters: Purely attention based local feature integration for video classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7834–7843.

[13] F. Sener, D. Singhania, and A. Yao, "Temporal aggregate representations for long-range video understanding," in *European Conference on Computer Vision*. Springer, 2020, pp. 154–171.

[14] S. Becker, R. Hug, W. Hubner, and M. Arens, "Red: A simple but effective baseline predictor for the trajnet benchmark," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[15] C. Scholler, V. Aravantinos, F. Lay, and A. Knoll, "What the constant" velocity model can teach us about pedestrian motion prediction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1696–1703, 2020.

[16] G. J. Mendis, T. Randeny, J. Wei, and A. Madanayake, "Deep learning based doppler radar for micro uas detection and classification," in *MILCOM 2016-2016 IEEE Military Communications Conference*. IEEE, 2016, pp. 924–929.

[17] J. Drozdowicz, M. Wielgo, P. Samczynski, K. Kulpa, J. Krzonkalla, M. Mordzonek, M. Bryl, and Z. Jakielaszek, "35 ghz fmcw drone detection system," in *2016 17th International Radar Symposium (IRS)*. IEEE, 2016, pp. 1–4.

[18] S. R. Ganti and Y. Kim, "Implementation of detection and tracking mechanism for small uas," in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2016, pp. 1254–1260.

[19] B. H. Kim, D. Khan, C. Bohak, J. K. Kim, W. Choi, H. J. Lee, and M. Y. Kim, "Ladar data generation fused with virtual targets and visualization for small drone detection system," in *Technologies for Optical Countermeasures XV*, vol. 10797. International Society for Optics and Photonics, 2018, p. 107970I.

[20] M. Laurenzis, S. Hengy, A. Hommes, F. Kloeppel, A. Shoykhetbrod, T. Geibig, W. Johannes, P. Naz, and F. Christnacher, "Multi-sensor field trials for detection and tracking of multiple small unmanned aerial vehicles flying at low altitude," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVI*, vol. 10200. International Society for Optics and Photonics, 2017, p. 102001A.

[21] A. Hommes, A. Shoykhetbrod, D. Noetel, S. Stanko, M. Laurenzis, S. Hengy, and F. Christnacher, "Detection of acoustic, electro-optical and radar signatures of small unmanned aerial vehicles," in *Target and Background Signatures II*, vol. 9997. International Society for Optics and Photonics, 2016, p. 999701.

[22] L. Hauzenberger and E. Holmberg Ohlsson, "Drone detection using audio analysis," 2015.

[23] S. Y. Nam and G. P. Joshi, "Unmanned aerial vehicle localization using distributed sensors," *International Journal of Distributed Sensor Networks*, vol. 13, no. 9, p. 1550147717732920, 2017.

[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[25] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1701–1708.

[26] C. Aker and S. Kalkan, "Using deep networks for drone detection," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[27] A. Schumann, L. Sommer, J. Klatte, T. Schuchert, and J. Beyerer, "Deep cross-domain flying object classification for robust uav detection," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[28] M. Saqib, S. D. Khan, N. Sharma, and M. Blumenstein, "A study on detecting drones using deep convolutional neural networks," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–5.

[29] C. Vondrick, H. Pirsiavash, and A. Torralba, "Anticipating visual representations from unlabeled video," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 98–106.

[30] Y. Wu, L. Zhu, X. Wang, Y. Yang, and F. Wu, "Learning to anticipate egocentric actions by imagination," *IEEE Transactions on Image Processing*, vol. 30, pp. 1143–1152, 2020.

[31] T. Nagarajan, Y. Li, C. Feichtenhofer, and K. Grauman, "Ego-topo: Environment affordances from egocentric video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 163–172.

[32] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Predicting the future: A jointly learnt model for action anticipation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5562–5571.

[33] J. Gao, Z. Yang, and R. Nevatia, "Red: Reinforced encoder-decoder networks for action anticipation," *arXiv preprint arXiv:1707.04818*, 2017.

[34] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7464–7473.

[35] Q. Wang, B. Li, T. Xiao, J. Zhu, C. Li, D. F. Wong, and L. S. Chao, "Learning deep transformer models for machine translation," *arXiv preprint arXiv:1906.01787*, 2019.

[36] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou, "Training data-efficient image transformers & distillation´

through attention," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10347–10357.

[37] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," 2020.

[38] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 244–253.

[39] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[40] D. Neimark, O. Bar, M. Zohar, and D. Asselmann, "Video transformer network," *arXiv preprint arXiv:2102.00719*, 2021.

[41] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price *et al.*, "Scaling egocentric vision: The epic-kitchens dataset," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 720–736.

[42] R. Kruse, C. Borgelt, F. Klawonn, C. Moewes, M. Steinbrecher, and P. Held, "Multi-layer perceptrons," in *Computational Intelligence*. Springer, 2013, pp. 47–81.

[43] A. Baevski and M. Auli, "Adaptive input representations for neural language modeling," *arXiv preprint arXiv:1809.10853*, 2018.

[44] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," *arXiv preprint arXiv:2102.12122*, 2021.

[45] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," 2020.

[46] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.

[47] J. Li, J. Murray, D. Ismaili, K. Schindler, and C. Albl, "Reconstruction of 3d flight trajectories from ad-hoc camera networks," 2020.

Information Preserving Frame-based Image Interpretation

Vasudeva Kilaru
Lovely Professional University
(Jalandhar)
Venus.kilaru@gmail.com

Abstract— Interpreting image data to neural networks is challenging. Deep convolutional neural network methods have shown promising results in interpreting image data to neural networks, with convolution and pooling operations over the traditional fully connected dense layers. The performance of these deep convolutional methods, however, is often compromised by the constraint that the convolution and pooling operations interpret the image data to neural networks by compressing the data into lower dimensions that lead to information bottleneck. To mitigate this, neural networks tend to be larger with more parameters (filters) thus increasing the computational cost (GPU Resources). In this paper, I present an information-preserving way to interpret image data to a convolutional neural net without any information bottleneck and with relatively fewer parameters, that also ensures no loss in information due to convolution or pooling operations. Uniquely my method adds two additional operations, one over the first regular convolution operation and the other operation next to the deeper convolution operation in the neural net. The first operation is to divide the image into individual frames by frame-based crop operation and then apply regular convolution, pooling operations to interpret individual frames of the image into low dimensional tensors that preserve information from being bottlenecked since operations use frames of the image instead of using total image data. The second operation is a convolution operation applied after joining all low dimensional tenors of individual frames to extract information that later passes through the rest of the layers. This idea of using frames can better help model tasks like image generation and completion as well. Using frames of data instead of the total image helps in parallelizing computations thereby drastically decreasing the computational cost and depth of a neural network. Experiments conducted on inception networks worked better with relatively small network architecture on vision tasks.

Keywords—image interpretation, compute cost, network complexity, feature detection, network parameters, task parallelization.

I. INTRODUCTION

Interpreting unstructured data, especially image data to artificial neural networks is challenging. Existing methods like convolutional neural networks interpret image data more efficiently than earlier methods of fully connected layers that lack **parameter sharing** and **sparsity of data**. Convolutional neural networks have shown promising results in the field of computer vision from image recognition to self-driving cars however; the static, rigid operations like convolution, pooling leads to information bottleneck that compresses data and even force the network to be large with increased parameters to perform well on tasks. In convolutional neural networks, static set of weights/filters are applied in common on total image to detect different features of different regions on the image through convolution or pool operations which lead to the higher number of less insight full convolution/pool operations on the image thereby increasing the number of operations/computations on an image to detect all features. Convolutional neural networks often encounter information bottlenecks due to higher static convolution and pooling operations so, to perform better on vision applications these networks require more hardware resources for computations.

In this paper, I present a frame-based crop operation, a method applied before and after the extreme convolution operations in a Convolutional Neural Network (CNN) architecture. The frame-based crop operation parallelizes operations and addresses issues like information bottleneck, higher static convolution operations, and high computations (convolution operations).

This method of using additional frame-based crop operation is capable of effectively interpreting the data to neural nets without any information bottleneck so the network is relatively small. This method is also capable of reducing the number of overall filters needed to detect any feature of the image by dynamic filters which are specific to specific regions on the image without any extra or insightful filters on frames that don't detect any feature thereby decreasing the number of convolution operations (computations).

The two advantages of additional operation i.e., frame-based crop operation, are: (1) Better interpreting the data in a relatively smaller network can reduce computational cost, (2) Ability to parallelize the operations on individual frames reduces the computational cost.

II. RELATED WORK

A. Foundational works on image interpretation

In the early days of the deep learning era, the vision applications like image classification or semantic segmentation were carried out by neural networks with fully connected dense layers [1][2], however, these fully connected layers were not so good at interpreting unstructured data, especially image data which lead to poor performance on vision tasks. Around the mid-'90s in the paper "Convolutional Neural Network for Image, Speech, Time series" the idea of novel convolution operation was proposed which improved the performance of machine vision tasks like image classification [3], facial recognition and so on, by spatial interpretation of image data

to neural networks. The Convolutional operations were widely adapted in all the vision applications even now in self-driving cars, image-related tasks because of two main advantages of CNNs over the traditional fully connected layers, (1) **parameter sharing** that drastically reduced the number of parameters in the network and (2) **sparsity of data**. CNN's have shown very promising results by extending their intuition applicability from computer vision to NLP [4] (Natural Language Processing). Sometime later other methods were added to CNN's architecture to reduce the computations like Pooling [5], Batch Normalization. With these advancements, CNNs have shown better results on computer vision applications from traditional image classification, image semantic segmentation in (Figure 1) to more advanced tasks like facial recognition [6] [7] [Figure 2], check detection [8] [Figure 3], object detection [9] [Figure 4], and autonomous driving [10] [Figure 5].

B. Recent works on image interpretation

Around 2011, the research team at Microsoft has introduced the concept of residual blocks or skip connections in the RESNET paper that enabled to better interpret the data into deeper layers and train without problems like vanishing gradient/ exploding gradient [11]. The idea of 1x1 convolution operations i.e., network to network layers is adapted into inception network from RESNET paper.

The idea of Frame-based crop operation is inspired by the YOLO (you only look once) algorithm [12]. The idea of dividing the full-size image into smaller grids proposed in the paper is effective on tough tasks like real-time object detection in autonomous cars, however, the FBCO operation reduces the compute costs like 1x1 convolutions or bottleneck layers and also reduce network size drastically.

The idea of Frame-based crop operation is applied on the inception network of inception blocks which outputs the inception modules formed by stacking multiple convolution/ pooling operations (as shown in Figure 1) and large computational costs are managed by adding a bottleneck layer [13].

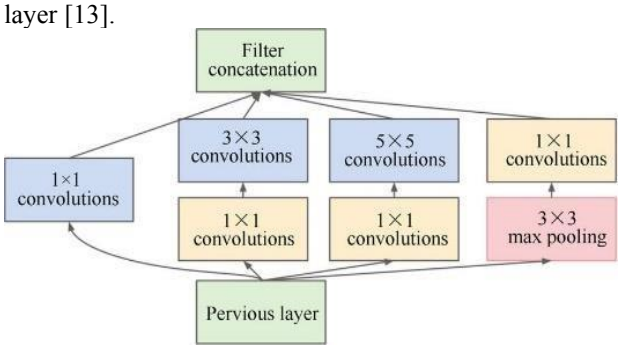


Figure 1. Inception block [13].

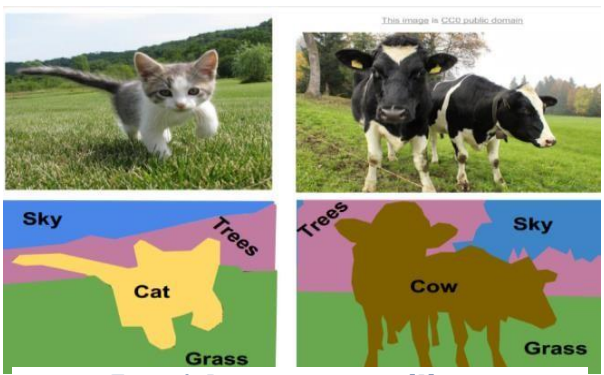


Figure 2. Image segmentation [5].

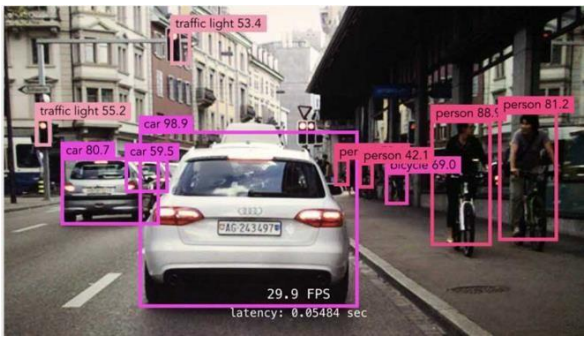


Figure 3. Object detection [8].

Figure 4. Facial detection and recognition [6]

Figure 5. Autonomous Car [9]

Despite having many proven applications, the neural network often needs a large network with high parameters (specifically filter weights that detect different features on image) to train well and perform well. The novel method proposed in the paper, which is inspired by the methods of tokenization of input in NLP, YOLO and Inception network can significantly reduce the size of the network along with the number of parameters in the neural network that enables us to train fast with low computational cost without compromising on the performance of on the vision tasks.

III. Information Preserving Frame-Based Operation

We first review how the conventional neural networks are applied to interpret image data for image classification problems (say).

Earlier images were interpreted in the form of a single-dimensional array that stores the pixel values (as shown in Figure 6) of the image [14], but this way of interpreting the image to neural networks had issues like large parameters (weights) and data sparsity. These issues lead to the need for large network models to be trained well on tasks and huge compute costs to perform computations of huge parameters associated with pixel values at each layer.

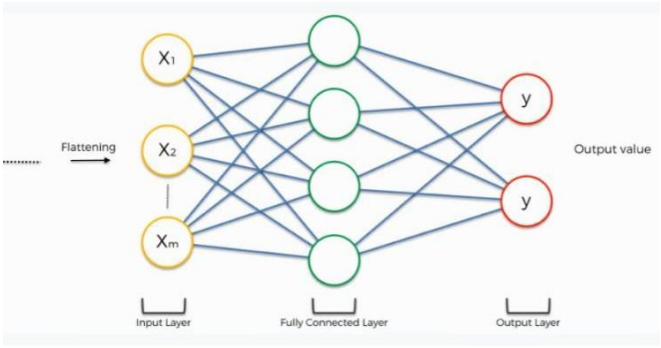


Figure 6. Fully connected layers for Image

In the LeNet paper-1989 by Yann Lecun, Convolution operations were proposed to do better image interpretation to neural networks than regular fully connected (dense) layers that need a relatively very small number of weights (parameters). Convolution operations (as shown in [Figure 7] [15]), which were inspired from the feature detection mechanism of human vision, worked out well due to the ability of parameters sharing and data sparsity that reduced the number of trainable parameters with concept convolution filters, used to detect features on the image.

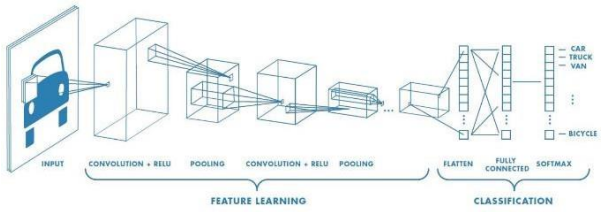


Figure 7. Convolutional layers for Image

Frame-Based Crop operation (FBCO)

The vision tasks are often tough for neural networks to generalize and be robust with fine variations because of their complex mapping function from input images to classes (say). Many advancements made neural networks work better in complex mapping vision tasks with larger networks. The computational costs are managed by bottleneck layers and backpropagation gradient issues are managed by skip connections or residual blocks. All the methods could significantly improve the neural networks to map complex functions but drastically increase the computational cost, the idea of FBCO is to decrease the computational cost without compromising on the performance on different vision tasks.

To avoid the information bottleneck of image data by conventional convolution, pooling operations and enable better information interpretation during training and testing with relatively low computational cost, I employ the frame-based crop operation. Below, I first describe the frame-based crop operation and then illustrate how it incorporates in the conventional deep convolutional neural network to improve image interpretation.

Frame-based Crop operation. As discussed in related work, existing methods of convolutional neural networks compress data to lower dimensions by bottlenecking the information (As a part of interpreting the image data to neural network). The main reason behind this is due to hardcoded convolution, pooling operations over the total image at stretch thereby increasing the number of parameters/filters, the computational cost to detect complete features. The frame-based crop operation, inspired from the method of tokenizing data, YOLO and Inception network with 1x1 convolution operations and bottleneck layers [13], proposed in this paper divides the single image data into individual dependent frames on which the conventional CNN methods are applied parallelly. As illustrated below and in Figure 8, the frame-based crop operation (FBCO) takes the image input and divides it accordingly, into individual image frames. These frames are later on passed through the regular convolutional neural network separately.

The actual execution of FBCO operation.

Original Image

11	21	32	22	25	76	87	99
22	62	19	49	61	65	43	76
10	08	20	47	55	77	21	43
54	06	57	12	32	57	13	25
21	66	40	15	54	42	44	54
19	54	21	36	87	41	43	33
23	23	71	31	37	98	53	21
73	99	66	23	34	76	73	35

Frame 1

00	**00**	**00**	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**	**00**	**00**
00	**00**	20	47	55	77	**00**	**00**
00	**00**	57	12	32	57	**00**	**00**
00	**00**	40	15	54	42	**00**	**00**
00	**00**	21	36	87	41	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**	**00**	**00**

Frame 2

00	**00**	**00**	**00**	**00**	**00**	**00**	**00**
00	62	19	49	61	65	43	**00**
00	08	20	47	55	77	21	**00**
00	06	57	**00**	**00**	57	13	**00**
00	66	40	**00**	**00**	42	44	**00**
00	54	21	36	87	41	43	**00**
00	23	71	31	37	98	53	**00**
00	**00**	**00**	**00**	**00**	**00**	**00**	**00**

Frame 3

11	21	32	22	25	76	87	99
22	62	19	49	61	65	43	76
10	08	**00**	**00**	**00**	**00**	21	43
54	06	**00**	**00**	**00**	**00**	13	25
21	66	**00**	**00**	**00**	**00**	44	54
19	54	**00**	**00**	**00**	**00**	43	33
23	23	71	31	37	98	53	21
73	99	66	23	34	76	73	35

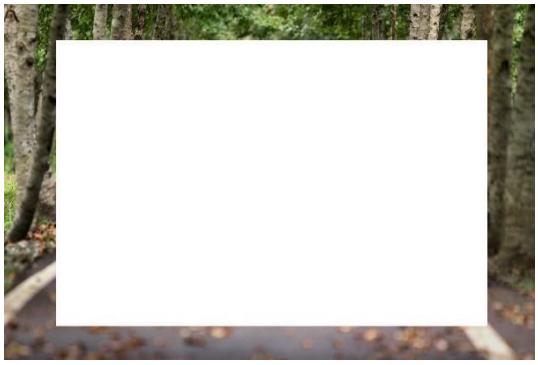
High-level intuition of FBCO operation.

Figure 8. Original Image (frame {0, 1, 2, 3})

Frame 1

Frame 2

Frame 3



A. FBCO-Parallelizing the operations.

The frame-based crop operation which divides the original image into individual frames as shown in Figure 8, enables us to parallelize the computations. Earlier the total image is passed into the CNN, but here the image is divided into frames that can pass through the CNN simultaneously resulting in a corresponding low dimensional representation. These representations are combined back to represent the total image in low dimensions. At least one convolution operation is performed before flattening the image data as it improves the quality of representing the image (as shown in Figure 9) in low dimensions. Thus, the final output can be used to perform many tasks related to computer vision applications.

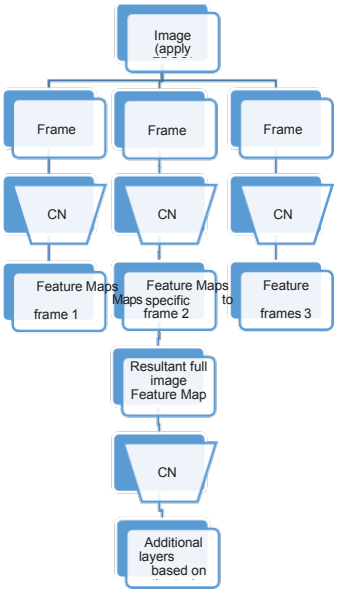


Figure 9. FBCO introduced CNN architecture

B. FBCO-Dynamic parameters on frames

The additional method of frame-based crop operation can significantly reduce the number of overall parameters thereby reducing the computational cost. The advantage of tokenizing the data i.e., dividing the total image into frames enables the model to learn specific filter weights to detect features on different regions or frames of the image that contribute to decreasing the number of filters to detect total features. For example, To detect all features of an image, let's suppose 256 unique filters are needed but practically 256 filters don't detect features in every region because, there may specific features like eyes or eyebrows that exists only in a particular region of the image which means the eye feature detecting filters are not useful to apply on the other parts of the image so, if we divide the image into frames, the model learns specific features on every region i.e., the model learns eye detecting filters only to that particular region/frame that has eyes thereby reducing the number of convolution operations of various filters throughout the image that decreases the computational cost.

C. FBCO-Feature Detection

Feature detection with FBCO remains as appropriate as it is in conventional CNN, however, FBCO can make feature detection much more effective with the concept of frames. In the conventional methods, feature detection takes place on the full image but with the FBCO, the full image is divided into individual frames over which the feature detection is done. As discussed in point C, it helps to reduce the number of parameters

by allotting only necessary and dynamic filters for detecting accurate features on each frame. For example:

01	01	01	01	01	01
01	01	01	01	01	01
01	01	01	01	01	01
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**

The above tensor is corresponding to a horizontal edge image and applies conventional convolution operation to detect edge over the image of the horizontal edge using the horizontal edge detecting filter, the output would be a low dimensional feature map same as the resultant feature map shown below.

Horizontal edge filter

01	01	01
00	00	00
-1	-1	-1

Resultant Feature map

00	00	00	00
03	01	-1	03
03	01	-1	03
00	00	00	00

and applying the same horizontal edge detecting filter on the image by the FBCO method also results in the same feature map along with an option to parallelize the computations faster on combining.

FBCO operation on horizontal edge image into frames

Frame 1

00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	01	01	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**

Frame 2

00	**00**	**00**	**00**	**00**	**00**
00	01	01	01	01	**00**
00	01	**00**	**00**	01	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**

Frame 3

01	01	01	01	01	01
01	**00**	**00**	**00**	**00**	01
01	**00**	**00**	**00**	**00**	01
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**
00	**00**	**00**	**00**	**00**	**00**

Resultant Feature Map

00	00	00	00
03	01	-1	03
03	01	-1	03
00	00	00	00

Therefore, by dividing the image into frames, operations like convolution and pooling operations can better interpret image data while without compromising on the efficient feature detection task that helps in performing better in a long run.

IV. Implementation Details

Step by step approaches to implement FBCO operations over the conventional convolutional neural networks are as follows:

1. The numerical tensor of the image is to be divided into frames via an operation called frame-based crop operation. The operation/function takes in the image tensor along with any two hyperparameters among the following:
    1. Stride-skip: The numerical value that specifies the method of cropping i.e., it tells, how many pixels need to be skipped while dividing the image into frames.
    2.Frames: The numerical value that specifies the total number of frames need to be made out of a full-size image.
    3. Shared pixels: The numerical value that specifies the number of pixels to be shared between every consecutive frame ensures the spatial connections between the frames.

The relation between all the three parameters with the full size of an image i.e., the number of pixels of the image is given by

(1) (Image is expected to be of shape nxn where n ∈ positive even integers).

$$f = [n/((s\_s - s\_p) + 1)] \quad (1)$$

the f-number of frames, n-shape of the image, s_s-number of strides to skip ∈ positive even integers, s_p number of shared parameters ∈ positive even integers.

Note-a. The FBCO operation, relations, equations are only applicable on images of shape nxn where n ∈ positive even integers.

b. The FBCO operation can be applied on images of any shape by applying a composition preserving multi-level spatial pooling layers to bring down images to shape nxn without losing any aesthetic value of image before FBCO [16].

2. Each frame is passed through a state of art convolutional neural network separately like VGG16 [17], Resnet [11], etc. The output of the neural network of each frame (low dimensional representation of each frame on applying operations like convolution, pooling) is then combined in reverse order to form an overall low dimensional tensor representing the full image. The combined tensor is passed at least through one convolution operation before flattening to ensure the data sparsity issue as shown in Figure 9.

3. The tensors are flattened into single-dimensional arrays to pass through fully connected layers to perform any task

related to vision applications from image classification, semantic segmentation to object detection, path detection.

A. Experiments

The addition of the FBCO method drastically reduces the computational cost and accomplishes any task with comparatively a smaller network, without compromising the performance. The method is applied to an existing state of art vision model inception. The experiments conducted have shown better performance on image classification tasks with a smaller network of inception models. The regular inception model has got an accuracy of 91.813% on dog breed classification at the lowest SoftMax. The FBCO applied inception managed to get the accuracy of 91.809% at third side branch SoftMax from the lowest and even more accuracy ranging 92%-95% at later SoftMax side branches in the inception network and with relatively reduced computations. The FBCO applied inception could do the task in a relatively lesser time (0.7x) than the regular inception on image classification with the reduced number of effective weights and by parallelizing the execution.

V. Conclusion

This paper presents a frame-based crop operation over the extreme convolution operations in any state of art CNN model to preserve essential information/features in an image. Interpreting the image data in low dimensional tensors by compressing it may lead to information bottleneck issues but with FBCO, the model ensures the best quality of image data interpretation which significantly reduces the network size, parameters and operations required to perform on any vision task without compromising on any factor. First, the frame-based crop operation is implemented on full image to break it down to smaller frames which enables to parallelize the computations and decrease the parameters thereby decreasing the computational cost. Finally, the resultant low dimensional tensors of each frame are combined and passed to layers to carry out machine vision tasks like object detection etc. The positive aspect of the method is that it significantly reduces the computational cost without any tradeoff with the performance of the model.

Acknowledgment

This research was supported by Naustone Inc, Lovely Professional University. I am grateful to the CEO (chief executive officer) of Naustone Mr Sreekar Nayani for all support both morally and intellectually to pursue research and bring out great insights on the subfield of vision i.e., unstructured data interpretation to deep neural networks.

Thanks to all the open-source research papers, lectures especially for IEEE CVPR papers and Coursera.org on various concepts that supported me in pursuing this research on data (image) interpretation.

References

[1] J. Schmidhuber. Deep learning in neural networks: An overview. Neural Networks, 2015. Published online 2014; based on TR arXiv:1404.7828 [cs.NE]. 2

[2] Ken-yuh hsu, Hsin-yu li, Demtri. Implementation of a Fully Connected Neural Network. IEEE Xplore. 2

[3] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. Neural Comput., 1(4):541–551, December 1989. 2

[4] Aditya Mogadala, Marimuthu Kalimuthu, Dietrich Klakow. Trends in integration of Vision and Language Research. ArXiv.org. 2, 3

[5] Scherer, Dominik & Müller, Andreas & Behnke, Sven. (2010). Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. 92-101. 10.1007/978-3-642-15825-4_10. 2

[6] Image is accessed on 02/12/2021 from https://tariq-hasan.github.io/concepts/computer-vision-semantic-segmentation/. Image of semantic segmentation. 2, 3

[7] Image is accessed on 02/12/2021 from https://sciforce.solutions/. Image of facial recognition and detection. 2, 3

[8] The Concept has been accessed from the below Wikipedia page https://en.wikipedia.org/wiki/Signature_recognition on 02/12/2021. 2

[9] Image is accessed on 02/12/2021 from https://www.analyticsvidhya.com/ . image of object detection in self-driving cars. 2, 3

[10] Image is accessed on 02/12/2021 from https://www.analyticsvidhya.com/ . image of object detection in self-driving cars. 2, 3

[11] Kaiming He, Xiangyu Zhang, Jain Sun. Deep residual learning for image recognition. ArXiv.org. 2

[12] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. You Only Look Once. CVPR. 6

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. Going Deeper with convolutions (Inception network). 1409.4842, September 2014. 2, 3, 6

[14] Image is accessed on 02/12/2021 from https://www.superdatascience.com/blogs/convolutional-neural-networks-CNN-step-4-full-connection . Fully connected layer representing image data.

[15] Image is accessed on 02/12/2021 from www.towardsdatascience.com. Convolutional Neural Network.

[16] Long Mai, Hailin Jin, Feng Liu. Composition Preserving Deep photo aesthetic assessment. CVPR 2016. 5

[17] Karen simonyan, Andrew Z. Very Deep Convolutional Network for Large Scale Image Recognition (VGG16). arXiv:1409.1556v6 **[cs.CV]** 5

Introducing Inclusive Construct Label-Centric Approach for Model Performance Enhancement in Autonomous Vehicles

Yashaswini Viswanath
RACE, Reva University
yashaswini.cse@gmail.com

Sudha Jamthe
Stanford University CSP
sujamthe@businessschoolofai.com

Suresh Lokiah
Independent Researcher
Sureshlokiah@gmail.com

***Abstract*—Driver Monitoring Systems (DMS) track movement of people using a camera inside the vehicle using AI to predict driver alertness to decide the safety of the driver and people on the road. Cameras collect huge amounts of data in all light conditions and activities of people inside the car. This data carries a wealth of insights about driver movement. Hence we propose a new label centric-approach by labeling the camera data with inclusive AI constructs for a more expansive annotation of the same dataset instead of the typical model-centric approach or data-centric approach to improve the performance of this AI. We used the DMD multi-model dataset for driver monitoring scenarios which comes with labeling movement to track 8 actions of the human texting with left or right hand, talking, drinking, phone call with left or right hand, reaching the side of the car or combing hair.**

We developed a binary classification CNN model for movement in the car. We tested the model trained against an inclusive AI constructed labeling option on the same dataset where we expanded the labeling to track movement of hair flying, scarf fluttering, hand waving or rubbing eyes. The results showed that the inclusive AI construct improved model performance without any change to the model algorithm tuning. Hence we recommend using a label-centric approach to improve labeling of data from camera streams such as the autonomous vehicle to be inclusive on an expanded construct for labels covering all people of all cultures in all lights, all hair, dress, and actions so AI model performance can be improved by capturing more knowledge by being more inclusive of all humans and their actions inside the vehicle.

Keywords—labeling, annotation, inclusive ai, construct, model performance, label-centric, autonomous vehicles, DMS, Driver Monitoring, Driver Attention, Driver Distraction

I. INTRODUCTION

Driver Management Systems (DMS) track movement of people using a camera inside the vehicle and AI to predict driver alertness from this data. Tesla tracks driver's alertness[1][1] in full self-driving mode in the vehicle to decide on safe driver rating for insurance premiums and to allow Full Self Driving capability access to Tesla drivers. Data has more knowledge than is captured by labels available in the training data and if we capture all the knowledge contained in the data, we can train the AI better. AI understands the data from the annotations from the labeling of the data. Labeling is typically outsourced to someone who does annotations without any knowledge of AI. Little thought is given to how bias can be introduced in models at the labeling stage by having a narrow definition of constructs of what the AI is modeled to predict. So our approach in this paper is a label centric-approach. All AI models in the vehicle have to be inclusive of all people to predict the movement and alertness of humans for the driver management system to recognize them as being alert to give control to save human lives inside the car and on the road.

This paper postulates that we can improve the AI model performance by improving labeling of movement with inclusive AI constructs for a more expansive annotation of the same dataset.

OEMs are using AI in the car to track driver alertness as an ADAS driver assistance feature to allow for the car to handover control to the human in case of the failure of the autonomous capability of the car. The success of this AI depends on its accuracy and AI model performance. With AI today, data scientists take one of two approaches to improve the model performance. They take a model centric approach and focus on improving the model with hyper parameter tuning or take a data centric approach and retrain the model with more training data.

A. Modeling Approaches of Today

Today ML model performance is improved using two approaches. One is to focus on improving model performance called a model-centric approach. Another is a data-centric approach to focus on the data to train the model with improved data that is reflective of the problem statement. Both of these approaches leave behind valuable knowledge in the data because of gaps in labeling that is not inclusive of all kinds of people and situations that represent an unbiased dataset. Hence we propose the label-centric approach here.

B. Labeling Approaches and their limitations

Today Labeling is done by a 3rd party company that does not involve the data scientist and most labels are created with elementary and simple constructs of actions. This leaves behind valuable information in the data that is not labeled and hence the value is lost as introduced by Inclusive AI Movement Construct by Susanna Raj in AI Ethics: Responsible AI And Inclusive AI Masterclass at Business School of AI. [2][2]

[1] Simon Alvarez, Tesla introduces Safety Score (Beta) system that incentivizes safe driving, www.teslarati.com, https://www.teslarati.com/tesla-autopilot-safety-scores-explained-fsd-beta/, (accessed on 28 Dec 2021)

[2] Business School of AI, (2021), AI Ethics: Responsible AI And Inclusive AI Masterclass by Susanna Raj [Online], https://businessschoolofai.teachable.com/admin/courses/1441962/curriculum/lectures/34275444

Hence we need a move exhaustive definition of constructs to define classes of labels to maximum value from the data. We call this as "Inclusive AI Construct" and have created a new Inclusive AV dataset for which we share the details below.

C. Introduction to Inclusive AI Construct in Labeling

Data is the new oil is a cliche that is in use today implying that data carries value to create AI to solve problems that were previously possible before AI. Raw data that carries information in it to create value is really not valuable if it cannot be used to train the AI. This is where we see the power of annotations to label all the knowledge in the data. Data that is labeled inclusively to gleam maximum insight from it is really valuable data. This Inclusiveness can be expanded by adding labeling constructs covering all races, genders, cultures, countries and lived experiences of all people who will engage with the AI.

II. Experiment Setup

A. Dataset Used For This Research

We used the DMD Driving monitoring multi-modal dataset[3][3] for driver monitoring scenarios. This database comes videos of driver driving from real and simulated environments showcasing a variety of driver distraction actions.

The DMD multi-model dataset for driver monitoring scenarios which comes with labeling movement and we used 8 actions of the human texting with left or right hand, talking, drinking, phone call with left or right hand, reaching the side of the car or combing hair in this research.

We chose to use actions that define movement as a proxy of distraction and hence our paper is focused on tracking the labeling construct of movement and expanding it to be inclusive. We tracked 8 actions of the human texting with left or right hand, talking, drinking, phone call with left or right hand, reaching the side of the car or combing hair.

We got 16000 frames of one video from the DMD Driver Model Dataset. We labeled them as movement and no_movement based on 8 actions and expanded them to 6 more actions that define movement inside a vehicle. See Table 1 shows schematic of the list of movement actions and what was included in the base labeled dataset and what was in the expanded inclusive construct dataset.

Baseline Labeled Dataset: We used one video with 8 simulated driver actions from the DMS Dataset as our baseline labeled data. We generated 16000 frames from this video and used this as our baseline labeled dataset.

See below for sample image frames showing the 8 actions of drivers from the DMS dataset showing driver actions of drinking water, fidgeting phone, etc.

Figure 2: Driver Movement Action act_driver_actions/phonecall_ left from the DMS dataset

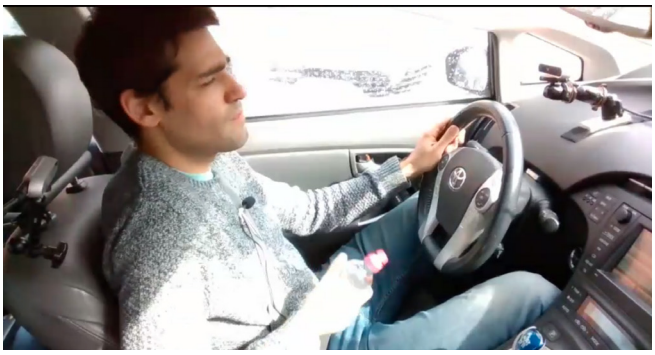


Figure 3: Driver movement Action act_driver_actions/drinking from the DMS dataset

Figure 3: Driver no movement action act_gaze_on_road/looking_road from the DMS dataset

Inclusive Construct Labeled Dataset:

We created a new dataset called Inclusive AV Dataset [4][4].

We made this similar to the DMD dataset to augment it with additional driver actions for movement inside a car or autonomous vehicle. This included 6 additional actions to

[3] Ortega, J., Kose, N., Cañas, P., Chao, M.a., Unnervik, A., Nieto, M., Otaegui, O., & Salgado, L. (2020). DMD: A Large-Scale Multi-Modal Driver Monitoring Dataset for Attention and Alertness Analysis. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops (Accepted).

show a woman with a scarf flying, long hair waving in the wind, scratching face, clapping, talking in road rage and hand movement that shows up as a blur. This expands the DMS dataset to be more inclusive of more types of people, outfits and situations inside the vehicle. Our goal with this dataset is to showcase more inclusive constructs to define human movement inside the vehicle. This is not an exhaustive dataset and can be expanded to more diverse people and actions in the future.

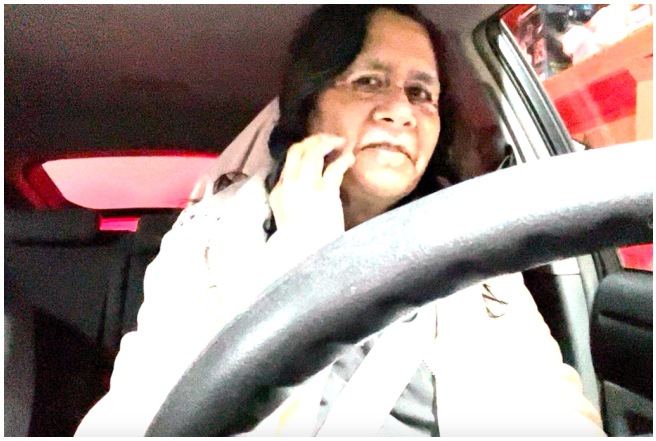


Figure 4: Driver movement action scratching from the Inclusive AI dataset

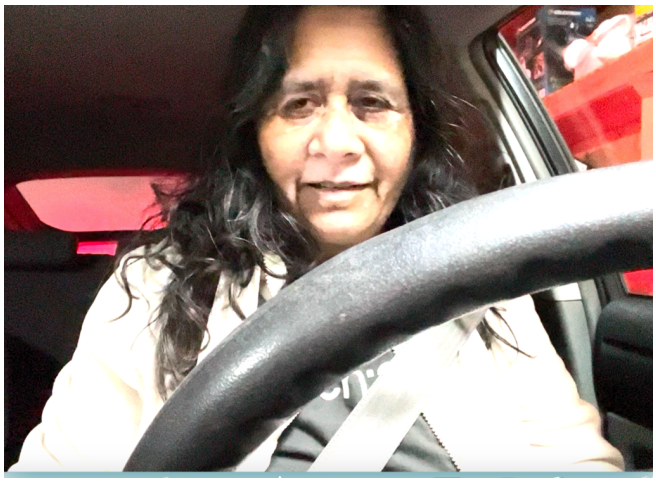


Figure 5: Driver movement action hair_flowing from the Inclusive AI dataset to be inclusive of people in the car with hair flowing

Figure 6: Driver movement action scarf_flying distracting driver attention as a movement from the Inclusive AI dataset

The Two Datasets Used in the experiments:

We added the Inclusive AV Dataset to one of 16K frames from gA_1_s4 video file from DMS dataset to create the Inclusive Construct Labeled Dataset.

So now we have a baseline labeled dataset with 8 actions defining movement class and rest as no_movement class and an Inclusive Construct Labeled Dataset with additional 6 actions that defines movement class and rest as no_movement class. We used this to compare the two datasets with the same model to test our hypothesis that label_centric approach can improve model performance.

The Actions Defining Movement class in the experiments:

	Movement Dataset Schema		
In baseline label dataset	***In inclusive construct labeled dataset***	***Action defining movement***	***Labeled Dataset***
n		act_gaze_on_road/looking_road	gA_1_s4_phase1
n		act_driver_actions/safe_drive	gA_1_s4_phase1
n		act_gaze_on_road/not_looking_road	gA_1_s4_phase1
n		act_hands_using_wheel/both	gA_1_s4_phase1
n		act_hands_using_wheel/only_left	gA_1_s4_phase1
n		act_driver_actions/change_gear	gA_1_s4_phase1
n		act_hand_on_gear/hand_on_gear	gA_1_s4_phase1
y	y	act_driver_actions/reach_side	gA_1_s4_phase1
y	y	act_driver_actions/hair_and_makeup	gA_1_s4_phase1
y	y	act_driver_actions/phonecall_right	gA_1_s4_phase1
y	y	act_driver_actions/phonecall_left	gA_1_s4_phase1
y	y	act_talking/talking	gA_1_s4_phase1
y	y	act_driver_actions/texting_right	gA_1_s4_phase1
y	y	act_driver_actions/phone	gA_1_s4_phase1
y	y	act_driver_actions/drinking	gA_1_s4_phase1
n		act_driver_actions/unclassified	gA_1_s4_phase1
n		act_hands_using_wheel/none	gA_1_s4_phase1
y	y	act_driver_actions/texting_left	gA_1_s4_phase1
n		act_hands_using_wheel/only_right	gA_1_s4_phase1
n	y	hand blur	Sudha_file_phase2
n	y	hair_flowing	Sudha_file_phase2
n	y	scarf_flying	Sudha_file_phase2
n	y	Scratching	Sudha_file_phase2
n	y	Clapping	Sudha_file_phase2
n	y	self talking road rage	Sudha_file_phase2

Table 1: Schema of actions defining movement class in the dataset used in the experiments.

B.Model Built for testing hypothesis

We tested the initial round of experiments using a pre-trained image classification model from teachable.withgoogle.com to train a model using two sets of labeled datasets and used 15 epochs, learning rate 0.01, batch size of 16. We used this as the baseline with the 8 actions and the other as an inclusive construct dataset with 14 actions defining movement inside a vehicle. Table 2 shows the initial results of the model developed using teachable.withgoogle.com.

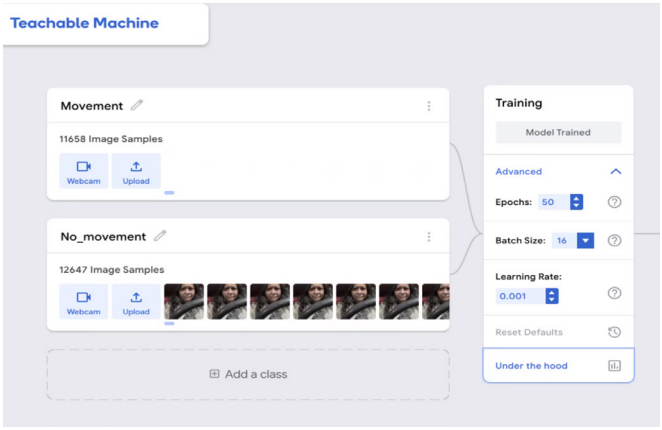


Table 2: Teachablemachine.withgoogle.com image classification classification

We decided to build our own CNN to validate the label-centric approach to model improvement for this research paper.

Convolution Neural Network based Image Classification Model

We developed image classification models using Convolution Neural Networks to use as the model that we will use in our experiments.

Model description:

We created a Convolution Neural Network model with three stages of Convolution Layers each with a max pooling layer. In order to avoid overfitting, the last max pool operation is appended with a dropout layer.This enabled us to create a model that learnt what is movement and what is no movement with trainable params of 553,314. In order to train the model we used DMD dataset and also augmented the dataset with frames captured in a custom setup. The resultant classification model was used for prediction on the test set of images.

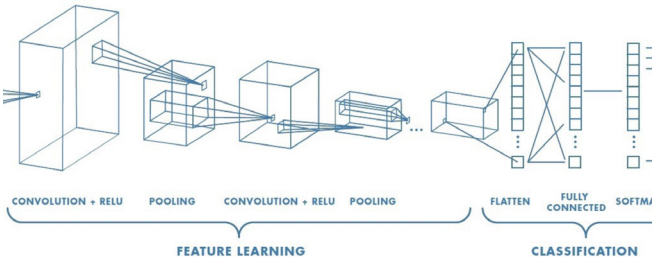


Image Credit: TowardsDataScience.com

C. Data Labeling Toolset

DMS Dataset comes in a VCD format with the annotations explained. It comes in JSON and is provided in VCD structure. It lists the objects and actions annotated by the labeler in each of the DMS Dataset videos. We needed the VCD structure to be flattened to csv formation to be loaded in Pandas dataframe. We needed to do this to summarize the driver actions from the annotation to create the movement and no_movement classes that we needed for the image classification. This helped with feature engineering to train our model. So we. built an open source tool for this research called as VCD Feature Enhancer which we have added to github as an open source contribution.[5][5]

III. Experiments

Our hypothesis that we experimented to prove was that a label_centric model will improve model performance. We decided to create a similar CNN model and the same dataset as fixed and ran experiments with the baseline labeled dataset of 8 actions defining movement and compared against the combined dataset of baseline labeled dataset and Inclusive Labeled Dataset of a total of 14 actions defining movement.

To do this, we developed the image classification model using ResNet and tested this using the Baseline Labeled Dataset video file gA_1_s4 which gave us 16K frames from the DMA Dataset. This dataset has 8 actions labeled and we tracked those as movement class and everything else as no_movement class.

A. Experiment Phase 0

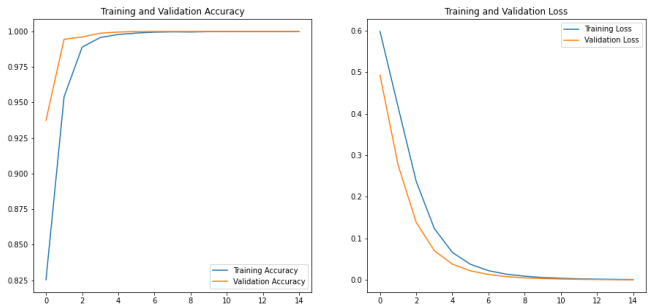
We did the first Phase 0 experiment to create a baseline model and test it out. To do this, we developed the image classification model using ResNet and tested this using the Baseline Labeled Dataset video file gA_1_s4 which gave us 16K frames from the DMA Dataset. This dataset has 8 actions labeled and we tracked those as movement class and everything else as no_movement class.

[5] VCD Flattener. (2022), Business School of AI, Accessed: Dec. 13, 2022, (Online), https://github.com/bsairesearch/DMD-Driver-Monitoring-Dataset/commits/master/vcd_flatten3.ipynb

Class	Model Performance		
	precision	***recall***	***f1-score***
Movement	0.79	0.92	0.85
No Movemnt	0.84	0.61	0.7

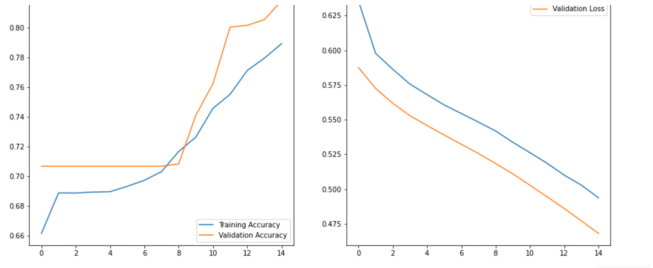
B. Experiment Phase 1

We did the first Phase 1 experiment to expand on experiment Phase 0 by repeating the Phase 0 experiment with the baseline labeled dataset and Inclusive AV Dataset but we did a baseline labeling of only the 8 actions defining movement. Table 2 for the schema of actions defining movement class in the dataset used in the experiments. We got the AI confusion matrix to measure model performance to use as baseline. We will use this to compare to the Experiment Phase 3 results to test our label_centric approach hypothesis of model improvement next.



C. Experiment Phase 2

We labeled the Inclusive AV Dataset to show the additional 6 actions to label a total of 14 actions defining movement. Then, we did the first Phase 2 experiment to expand on experiment Phase1 by repeating the Phase 1 experiment with the baseline labeled dataset and full labeled Inclusive AV Dataset with labeling of only all 14 actions defining movement. Table 1 shows the schema of actions defining movement class in the dataset used in the experiments.



Class	Model Performance		
	precision	***recall***	***f1-score***
Movement	0.79	0.92	0.85
No Movemnt	0.84	0.61	0.7

D. Experiment Results

We compared the accuracy of the CNN models trained from the baseline labeled dataset and the inclusive AI construct dataset. We found that the accuracy and precision improved through the inclusive labeling independent of model performance from improving the model by a data-centric approach, thus proving that the label_centric approach increased model performance.

As you can see, our goal here was not to improve the overall model performance. To that effect, we did not focus our experiments to improve model performance with a model_centric approach or data_centric approach. We focused only on a label_centric approach and saw improvement in the model performance when trained with a benchmark test dataset. So this leaves room to add incremental model performance by continuing to improve the model performance with model_centric and data_centric approaches additional to the label_centric approach demonstrated by our experiments.

IV. Call for Collaboration

Data is the new oil is a cliche that is in use today implying that data carries value to create AI to solve problems that were previously possible before AI. Raw data that carries information in it to create value is really not valuable if it cannot be used to train the AI. This is where we see the power of annotations to label all the knowledge in the data. Data that is labeled inclusively to gleam maximum insight from it is really valuable data.

In the research for this paper we propose a label-centric approach to model performance improvement. We focused on showing incremental performance improvement by improving labeling with better inclusive constructs. We would like to call for collaborators and future research to build upon our research because much work awaits us to benefit from this research. Future research can show whether model performance will improve if we augment label-centric performance improvements with existing model-centric and

date-centric approaches or whether these improvements are independent of each other.

We chose to focus on the camera data watching humans inside the car. Future research can find out if there are certain types of data or environments that support a data-centric approach to create model performance improvements over other approaches.

V. Conclusion

We are introducing a label_centric approach to model performance improvement. We found that the data inside cars and autonomous vehicles captures humans in several actions while only a limited set of actions are labeled and tracked to check for driver attention in Driver Management systems in cars today. This is not inclusive of all races, people, cultures, countries, outfits and movements that is reflective of all people who will be exposed to Driver Attention Systems flagged by the AI in the vehicle. We tested the model performance improvement by comparing the performance of a base model with a set of 8 actions defining movement inside a vehicle against 14 actions defining movement in the same camera data inside the vehicle. We found that we can improve the AI model performance by improving labeling of movement with inclusive AI constructs for a more expansive annotation of the same dataset.

This opens up a new label_centric model improvement approach to AI model performance improvement not limited to autonomous vehicles; the camera data inside the car calls for a more inclusive labeling covering many different people, situations, outfits and activities. We call for collaboration from future researchers to build upon our work to expand the label_centric approach to test for how it augments model_centric and data_centric approaches and in what kind of data situations it lends itself for optimal model performance.

We are very hopeful that with an inclusive construct labeling of data and improving models with a label_centric approach we will be able to uncover knowledge in data across the globe covering diverse set of people of all races, genders, cultures and life situations to make AI solve an even broader set of problems for all mankind.

References

[1] Simon Alvarez, Tesla introduces Safety Score (Beta) system that incentivizes safe driving, www.teslarati.com, https://www.teslarati.com/tesla-autopilot-safety-scores-explained-fsd-beta/, (accessed on 28 Dec 2021)

[2] Business School of AI, (2021),

AI Ethics: Responsible AI And Inclusive AI Masterclass [Online], https://businessschoolofai.teachable.com/admin/courses/1441962/curriculum/lectures/34275444

[3] Ortega, J., Kose, N., Cañas, P., Chao, M.a., Unnervik, A., Nieto, M., Otaegui, O., & Salgado, L. (2020). DMD: A Large-Scale Multi-Modal Driver Monitoring Dataset for Attention and Alertness Analysis. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops (Accepted).

[4] Inclusive AV Dataset, Business School of AI, 2022, [Online], Available: ttps://github.com/bsairesearch/inclusive-av-dataset

[5] VCD Flattener. (2022), Business School of AI, Accessed: Dec. 13, 2022, (Online), https://github.com/bsairesearch/DMD-Driver-Monitoring-Dataset/commits/master/vcd_flatten3.ipynb

[6] Teachable Machine, Google, Accessed: Dec. 28, 2022. (Online), https://teachablemachine.withgoogle.com/

Geo-contextual TV Consumption Patterns using Unsupervised Learning methods

Harshil Agrawal
Product Analyst,
MiQ Digital, Bangalore,
INDIA
harshil.agrawal@miqdigital.com

Nitin Vinayak Agrawal
Data Scientist I,
MiQ Digital, Bangalore,
INDIA
nitin.vinayak@miqdigital.com

Shubham Gupta
Team Lead,
MiQ Digital, Bangalore,
INDIA
shubhamgupta@miqdigital.com

Mrigank Shekhar
Data Scientist II,
MiQ Digital, Bangalore,
INDIA
mrigank.shekhar@miqdigital.com

***Abstract*—This In present times, with an ever-increasing television demand, advertisers are more and more interested in studying television content from the consumer's perspective. TV communities emerging due to this consumption, are crucial for digital marketing planners since it gives a clearer view into who the brands' audience could be, what they like, what they do. This would enable the planners to not only understand the avenues to reach them but also understand what creative messaging works best, what social messaging works best and eventually better returns. This paper examines and defines television consumers behavior to gain a more in-depth understanding of the target audience. As the Digital world is progressing towards a cookie less future i.e., cookie level information won't be available, we ought to look into aggregated data at the geo level. The unsupervised clustering experiments, (using KMeans), are performed on TV data provided by major TV players in the market. In this paper, we shall be walking you through the initial steps of data preparation and techniques used for clustering. In the later half, we will be discussing more on the techniques, experiments and the interpretation of TV personas with data**

Keywords— TV Consumption Behavior, online advertising, K-Means, unsupervised learning,

I. Background

Over the years researchers have tried to identify consumer behaviour based on TV viewership, e.g. [1] have used the two-way set-top box by capturing clickstreams of channel-changing behaviour when the audience uses a remote control handset to interact with the set-top box; [2] Analysis of audience interest and user clustering based on program tags. In this paper we attempt to identify TV consumption based clusters or personas of zip codes in the US, taking into account different features aggregated at zip code level like TV genre watch duration, weekday/weekend TV viewership using the TV data provided by data providers in the US. Our major focus throughout the analysis will be to arrive at stable and robust clusters with respect to time. We have also accounted for robustness while creating features to check the stability of clusters which will be discussed in further sections

II. Introduction

Historically, we have seen how technology has impacted the lives of people. With this fast-paced technological revolution, each individual has been a witness of its impact in one way or the other. With the support of high-speed internet and different tech-enabled devices, content watching has drastically increased. Those far to reach corners of the world, for which necessities seemed difficult, thanks to technology, are now watching TV shows at their leisure.

According to the Adobe Digital Insights Advertising Report [3], 74% of Americans feel that the television commercials they view are irrelevant to them. This is a huge proportion of the total audience, that also shows the number of capital advertisers are wasting on advertising to uninterested audiences. Creating TV personas will help target relevant audiences thus optimising the ad campaigns.

III. Objective

TV watching behaviours are essential for different businesses in a way that will help them to target advertisements to a specific audience segment at a particular time for their brand. In this paper, we will try to segment different audience behaviours by performing unsupervised clustering exercises on their TV watching patterns using different third party TV data providers across the US. TV data is collected via Automated Content Recognition(ACR) [4] where the content is recorded innately without intervention from the consumer. This ensures that every session recorded is accurate and is consistent across the dataset. This data is pre-processed and aggregated at a geo-contextual level for unsupervised learning to happen. Here we are clustering the geo level TV viewing features to understand if there are any naturally occurring communities when it comes to TV viewership. These communities are crucial for digital marketing planners since it gives a clearer view of who the advertisers' audience could be, what they like, what they do. It would also enable the digital marketing planners to not only understand the avenues to reach them but also understand what creative messaging works best, what social messaging works best.

From a business perspective, this will help better the storytelling and insights consumed. Digital marketing planners can include TV consumption clusters or let's say, personas, as packages in their plans.

This paper proposes that TV Personas that are created using an unsupervised learning approach in TV viewership majorly consist of the following three modules: TV Dataset Exploration, Unsupervised clustering, Cluster interpretation. In further sections, we will focus on the proposed methodology for the same.

IV. Methodology

A. TV Dataset Exploration

Our dataset provides the user's TV viewing logs. The data is provided by a third party vendor. Automatic Content Recognition(ACR) [4] is used for data collection from a user's TV set.

There are three types of features that are present in the dataset. Firstly, the data contains geo-information i.e. zip codes where audiences are present. Secondly, temporal features e.g. Time of the day, Day of the week when the user has watched the TV show. Thirdly, the type of content being watched mainly includes the title, description and genre of the show.

The dataset is present at session level (i.e. Activity recorded when a user starts watching a tv show until the user either switches to another show or the show ends). Since the records are related to each other on timestamps, it is essential to identify any time-based components present in the dataset that might impact the clustering exercise done for TV Personas.

B. Time Series Analysis

Time series analysis(TSA) [5] is the study of a sequence of data points (in our case TV viewership) collected over an interval of time. Since we are interested in Geo-level TV consumption patterns, first we aggregate our raw data at the zip code level with the feature as Mean watch duration per day at a given zip code. The mean watch duration can be inferred as a time series equation ,

$$Y_{TS} = \text{Trend} + \text{Seasonality} + \text{Residual} \qquad (1)$$

Figure 1 shows the time series decomposition of mean watch duration into its components. On the X-axis we have time in days (90 days), Y-axis is the value of time series, trend, seasonality and residual. It is evident from the figure that there is no clear trend in the time series as it is oscillating between 700 seconds and 550 seconds. Here we have checked for seasonality for a period of 7 days, an explicit repeating pattern is visible. This shows 7 days are periodic in the watch duration feature.

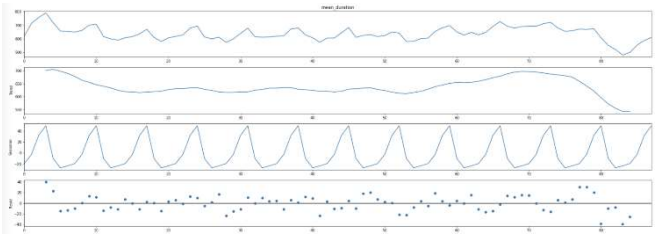


Figure 1: Time Series decomposition with periodicity = 7 days

Figure 2 shows time series decomposition with a period of 14 days, here also it is observed that 2 peak and valley patterns are repeating at an interval of 14 days. The 2 peak and valley pattern is somewhat symmetrical across 7 days. This confirms that there is a weekly seasonality present in the aggregated dataset. Hence, it is essential to incorporate features that are robust towards the weekly seasonality in our modelling exercise.

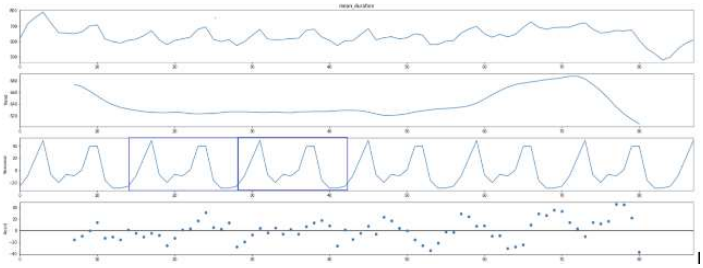


Figure 2: Time Series decomposition with periodicity = 14 days

C. Robust Feature Selection

One of the objectives was to make the tv persona stable and consistent with respect to time for this we performed preliminary analysis for the robust feature selection. In this analysis, we have created the features for different time intervals i.e 1 month, 2 months and 3 months and performed the correlation analysis [6] between the features. It can be inferred that the higher the correlation, the more robust the features are.Correlation can be denoted as :

$$r = \frac{N\sum xy - (\sum x)(\sum y)}{\sqrt{[N\sum x^2 - (\sum x)^2][N\sum y^2 - (\sum y)^2]}} \qquad (2)$$

where, $N = number\ of\ pairs\ of\ scores$

$\sum xy = sum\ of\ products\ of\ paired\ scores$

$\sum x = sum\ of\ x\ scores$

$\sum y = sum\ of\ y\ scores$

$\sum x^2 = sum\ of\ squared\ x\ scores$

$\sum y^2 = sum\ of\ squared\ y\ scores$

D. Noisy Data processing

Since our data is at the session level, there are noisy records present that do not represent the true TV watching behaviour, e.g. people switching between channels to reach their intended program. To remove these rows, a filter for minimum duration watch was applied. This would enable us to get more generalized data specific to TV consumption patterns devoid of any fluctuations.

V. EXPERIMENT

In this paper, we have tried out three different unsupervised learning methodologies for clustering the aggregated dataset and the creation of TV personas.

K-means clustering algorithm

The conventional k-means algorithm is described in this section. K-means clustering [7] is a data mining technique to group big sets of data. In accordance with this algorithm, k data points are selected as initial cluster centres, and distance is calculated between each centre and each data point. Points are assigned to their nearest centres, centroids

are recalculated and the process is repeated until the square error is within acceptable limits. Square error can be defined as :

$$E = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left|\left|x_{ij} - m_i\right|\right|^2, x_{ij} \qquad (3)$$

(x_{ij}) *is the sample* j *of the class* i
(m_i) *is the center of class* i

A. Experiment 5.1

Data is aggregated at zip code level for the following themes: genre, daypart, day of the week, content type (Linear network or OTT [8]) .Each theme consists of 5 measures which would ultimately define the robustness of the cluster. These measures are namely :

- daily average viewing time
- daily avg distinct sessions
- daily avg distinct shows watched
- daily avg distinct channels watched
- daily average viewing time per session

Total 310 features were created, which are capturing every aspect of the TV behaviour, but consisted of more cardinality in terms of features. Empty values were imputed with zeroes to fulfil the use case. The distribution observed for average watch time per session per day was observed gaussian. While for every other measure across themes, the distribution was observed as Poisson(Figure 3).

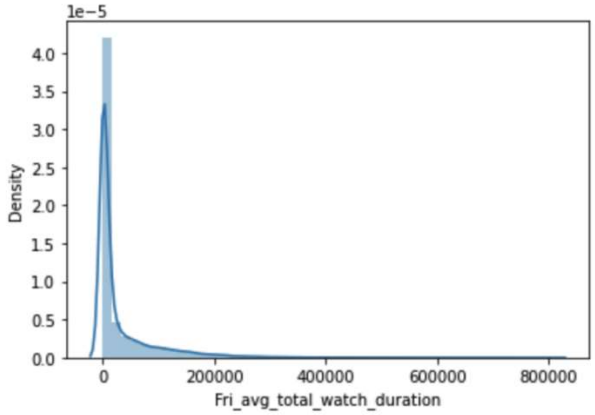


Figure 3: Avg total watch duration - Poisson Distribution

. Several pre-processing techniques like scaling, log normalisation, bin scaling were tried. Of all standardisation techniques, binning proved to be most effective. All the feature sets were binned in 20 bins. Bins were scaled between 0 to 1 as shown in figure 4.

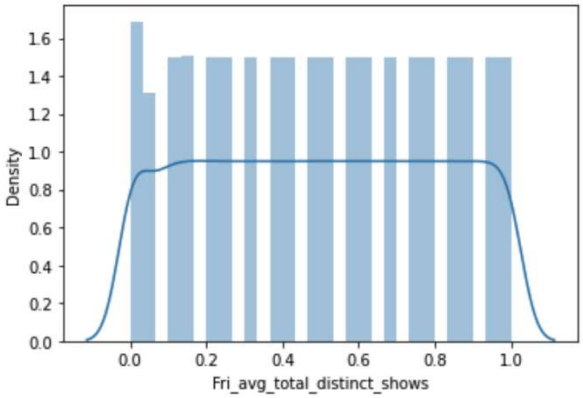


Figure 4: Bin Scaled feature

The correlation plot between features shows that a lot of variables out of 310 features are correlated. We decided to not reduce the columns via correlation or VIF as it might lead to the loss of some data. Instead, PCA [9] is done to identify 95% variance from 310 features.

Selected PCA [9] features are given to the K Means algorithm to perform clustering. Different iterations are tried out for a selection of an optimal number of clusters with the help of a silhouette visualiser [10].

B. Experiment 5.2

Data is aggregated at zip level along with the following themes: Genres, Daypart, content type, Weekday-Weekend (derived from Day of Week). Each theme consists of a single measure, i.e., total viewing time per television set-top box over 90 days in hours. Total 58 features are obtained from the dataset, out of these 10 features are static in nature i.e., these features would be prominent e.g., Weekday-Weekend, content-type, daypart. The other 48 features are dynamic in nature i.e., their labels might change with the time period selected for the dataset. To select robust and consistent dynamic features we perform correlation analysis as mentioned in the previous section. 43 features are selected. Several preprocessing techniques e.g., Robust Scaling, standard scaling and bin scaling were tried. Bin scaling gave the best results in terms of cluster cardinality. Bin scaled features are given to KMeans for clustering. Different iterations are carried out to select the optimal number of clusters and maintain balanced cluster cardinality. Finally, 7 clusters are selected for TV consumption and Demographics labelling.

C. Experiment 5.3

In this method genre was the main theme and we have combined genre with the daypart to create the feature. We have used the Robust Feature Selection methodology to select the robust feature. After selecting the robust feature, we used dendrogram (Figure 5) i.e., visual representation of the compound correlation data to combine the similar features. To normalize the data, we did column scaling on the combined features. This data was given to the k means algorithm. Training the model resulted in 8 clusters with a silhouette score [10] of 0.30

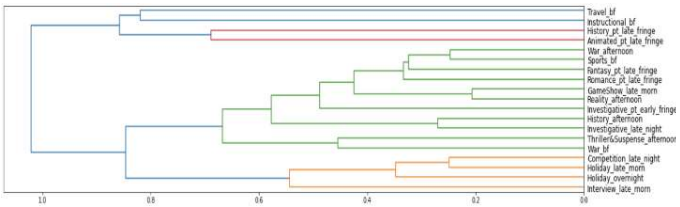


Figure 5: Dendrogram

VI. OBSERVATIONS

A. Evaluation Metrics

Silhouette visualiser[10] and cluster cardinality were considered while comparing the experiments.

Silhouette [10] is the measure of how similar the data points are to other data points from the same cluster. For each data point, the measure can be obtained in the range (-1, 1). The greater the value, the better the fit to the cluster to which it should be categorised and the lesser the fit to other clusters.

Cluster Cardinality is the number of data points in each cluster. Good clustering ensures that the data points are evenly distributed among clusters. Experiment 2 is selected for cluster labelling and further TV persona creation.

B. Labelling the TV Personas

To interpret the cluster we are using the census data and tv feature used for clustering. As the output of the clustering exercise is on zip code level and there is one to one mapping between the zip code and cluster. So we have joined the output and the census data using zip code as the primary key and then calculated the index using the below formula. The TV persona we arrived at is of 7 degrees. Two dimensional plot for the clusters obtained in Experiment 1 is shown in figure 6 and figure 7 shows the sample TV persona obtained at the end of experiment 2.

$$\text{Index} = \frac{(\text{No. of overlap users in the particular segment/total overlap users in all segments})}{(\text{Total users in segment/Total users in all segments})}$$

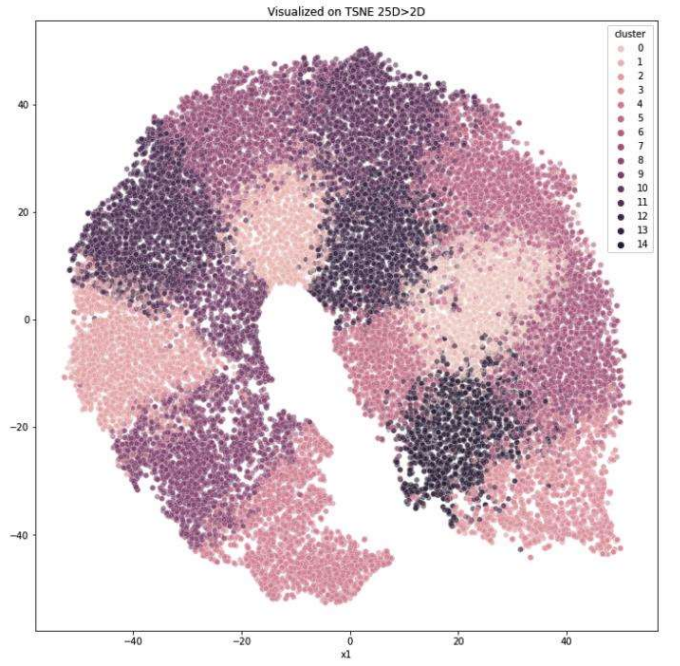


Figure 6: TSNE Plot of Clusters

TV Persona	TV Viewing Characteristics	Demographics	# of Zip Codes	Total Population
Scholars	1. Consumes mostly OTT content 2. Prefers watching content belonging to the following genres - Musical, Travel, Heath&Medicine, Home Shopping and Documentary relatively more in proportion than others 3. Watches TV likely during Early & Late Fringes 4. Likely to watch TV over weekends	Age - 15-24 Income - <$25k USD Gender - Male & Female Marital Status - Single	3856	35626

Figure 7: Sample TV Behaviour (TV Persona)

VII. CONCLUSION AND FUTURE WORK

Digital marketing planners are confronted with different marketing challenges like coming up with the best budget allocation across a variety of strategies like spending on Linear TV or OTT channel or display. The geo-contextual TV personas proposed in this paper serve as a guide to digital planners to plan their budget around how to spend their TV-related budget efficiently. In this paper, we have already discussed various techniques which help in producing a more robust set of features and eventually robust sets of clusters of TV consumption patterns. However, the research done was just to identify the geo-contextual TV consumption patterns with respect to digital planners. What warrants additional research is when to refresh the clusters to achieve the same robust definitions of clusters and how to handle macro events like COVID-19 in identifying the patterns which led to a drastic shift in the TV viewing behaviour in a single night.

REFERENCES

[1] Chang, R.M., Kauffman, R.J. and Son, I., 2012, August. Consumer micro-behavior and TV viewership patterns: data analytics for the two-way set-top box. In Proceedings of the 14th Annual International Conference on Electronic Commerce (pp. 272-273).

[2] Yin, F., Pan, X., Chai, J. and Zhang, W., 2016. Analysis of audience interest and user clustering based on program tags. Int. J. Hybrid Inf. Technol., 9(11), pp.79-90.

[3] Wolk, A., 2018. Television Is Embracing Audience Segmentation As Addressable OTT Continues To Explode. www.Forbes.com.

[4] Mittal, A. and Gupta, S., 2006. Automatic content-based retrieval and semantic classification of video content. International Journal on Digital Libraries, 6(1), pp.30-38.

[5] Cryer, J.D. and Chan, K.S., 2008. Time series analysis: with applications in R (Vol. 2). New York: Springer.

[6] Ratner, B., 2009. The correlation coefficient: Its values range between 1 -1, or do they?. Journal of targeting, measurement and analysis for marketing, 17(2), pp.139-142.

[7] Wang, J. and Su, X., 2011, May. An improved K-Means clustering algorithm. In 2011 IEEE 3rd international conference on communication software and networks (pp. 44-46). IEEE.

[8] Mendiratta, A., Wong, S., Grimm, R., Yogeshwar, J. and Archiquette, S., 2015, October. Big data analysis for effective monetization of over the top TV content. In SMPTE 2015 Annual Technical Conference and Exhibition (pp. 1-6). SMPTE.

[9] Sehgal, S., Singh, H., Agarwal, M. and Bhasker, V., 2014, November. Shantanu,“. In Data Analysis using Principal Component Analysis,” in 2014 International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom).

[10] Arabie, P., Baier, N.D., Critchley, C.F. and Keynes, M., 2006. Studies in classification, data analysis, and knowledge organization.

[11] Ma, Z., Yang, Y., Wang, F., Li, C. and Li, L., 2014, November. The SOM based improved k-means clustering collaborative filtering algorithm in TV recommendation system. In 2014 Second International Conference on Advanced Cloud and Big Data (pp. 288-295). IEEE

Machine Learning Implementations Scrutinized With A Process Re-engineering Lens

Abhinav Mathur
Sr Consultant, IQVIA
abhinav.mathur@iqvia.com

Sunny Verma
Principal Data Scientist, Thales
sunny.verma@thalesgroup.com

Arun Dahiya
Engagement Manager, IQVIA
arun.dahiya@iqvia.com

***Abstract* – Despite the advances in data acquisition, storage and algorithm capabilities, the impact of Machine Learning (ML) initiatives is sometimes debated across sponsoring units and organizations. While there is a possibility of the data and models not having the desired predictive value, there is a possibility that the desired business value is not realized due to an improper implementation of the interventions from the predictions intended. We discuss the chronological nature of data richness and the increase in predictive capability, and conversely there exists an inverse relationship between most effective interventions and the richness of data available for actionable predictions across process & product lifecycles. It thus becomes imperative to view ML implementations not as technology implementations but as Business Process Re-engineering initiatives to deliver the optimal business / process returns.**

Keywords—Machine Learning, business transformation, Business Process re-engineering, Strategic planning, Proof of Concept, Intelligent interventions, Analytics, Customer Life Cycle

I. INTRODUCTION

Despite the tremendous increase in compute capabilities, advent of cloud, sophistication of algorithms and the richness of data, Machine Learning (ML) implementations are debated in terms of impact towards of business objectives. This may beget the question that despite tremendous increases in components that enhance Machine Learning algorithms, the increase in richness and availability of data to train and improved accuracies, there could exist a possible disconnect between the measurement of the business objective and the model's accuracy metrics.

A significantly probable scenario is the mismatch between a business outcome and Machine Learning development. A model with good performance metrics is not a guarantor of a business success as good predictions must also have effective intervention, and the subsequent intervention must result in a measurable improvement of the business metric to have a successful Machine Learning implementation.

A top-down approach which considers a business metric to improve, based on the entity data is very similar to a Business Process Re-engineering (BPR) and thus Machine Learning implementations need to be considered as a Business Process Re-engineering implementations and worked on despite the traditional iterative nature of Machine Learning development.

This paper discusses on a broad methodology of structuring a Business Process Re-engineering leveraging Machine Learning and identifying, defining, and developing relevant criteria, interventions, and feedback mechanisms.

II. BUSINESS PROCESS RE-ENGINEERING LEVERAGING MACHINE LEARNING

Intended ML implementations are conceptualized for targeted interventions and the subsequent discovery phases are developed to improve upon a business metric. ML Proof of Concepts, (POCs) can thus follow a BPR process. The process would involve the following steps, as chronologically outlined below

(1) Quantification of improvement metric
(2) Data overview
(3) Intervention design
(4) Relationship between metric and model output
(5) POC Evaluation
(6) Production deployment / rollout

Quantification of improvement metric – For an effective improvement in a business, it is imperative to define the key performance metric in which an improvement is to be observed both between a base scenario established pre-rollout and the metric performance post POC or production rollout.

Data Overview – Data quality is directly correlated with time of the process / entity and thus for an effective prediction, the data quality as well as the time taken to validate predictions becomes paramount.

Intervention design – The model outputs are essentially insights upon which an intervention is required. An intervention however cannot improve the outcomes for an entire population and thus must be optimized.

Relationship between metric & model output – Once the model outputs are determined, a relationship between model accuracy and the improvement of the business metric must be established within the relevant confidence boundaries depending on the intervention post availability of insight.

POC Evaluation – An ML implementation and its subsequent intervention must first be evaluated in a pilot mode for a non-critical workload or a smaller sample to evaluate the improvement from the intervention and its extrapolation for a full-scale implementation.

Production deployment roll-out – Once the effectiveness has been established in a POC phase, the subsequent risks for a rollout must be evaluated with roll-back plans and extensively monitored with roll-back contingency for an established time.

III. CHRONOLOGICAL PROPERTY OF DATA QUALITY, MODEL ACCURACY AND INSIGHT INTERVENTION

As discussed in the earlier sections, for an effective implementation relies heavily on the effectiveness of the post insight intervention from the Machine Learning model. Thus, we can establish that there exists an interdependent relationship between the data quality, model accuracy and the

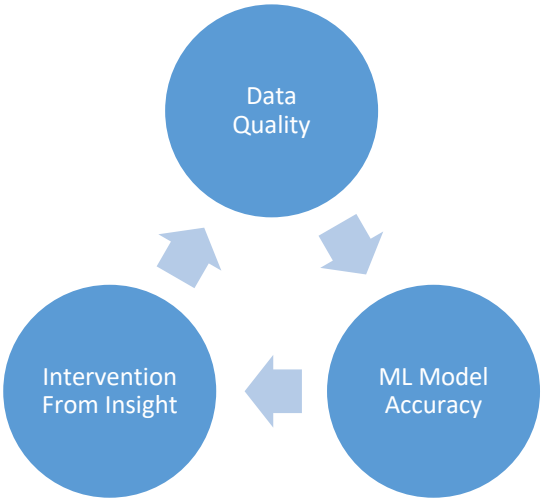


Fig.1 Inter-relationship between data quality, model accuracy and process intervention

insight intervention. A good data quality is more likely to result in a good accuracy and in turn a good accuracy would feed a more effective intervention which subsequently will improve the data quality in a chronological manner as represented in figure 1.

However, from the perspective of an entity life cycle we know that the data richness increases over a period, i.e. As an entity (either a product, person, customer, condition etc.) starts with very limited associated data and the data gets richer chronologically as more data is acquired.

This chronological property plays a very important role in identifying the stage at which a machine learning model is trained and expected to generate predictions.

Models trained on limited data could suffer from a poor training and thus not be accurate. This challenge can be compounded by the time taken to identify the dependent variable (if the Machine Learning algorithm is being trained for a supervised learning problem).

An example could be predicting customer churn post 6 months purely at first step of a customer lead with limited data. A model trained in this hypothetical scenario will have data with limited predictive capability and suffer from the time taken to validate the results in a POC / Production phase where it will take upwards of 6 months to validate the predictions of the model. This can be represented graphically as below in figure 2.

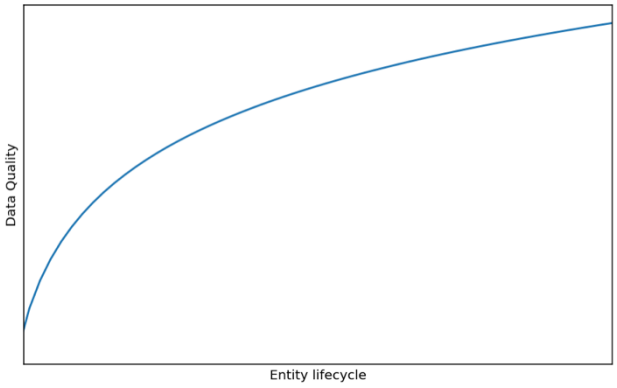


Fig.2 Summary representation of data completeness of an entity over entity life cycle

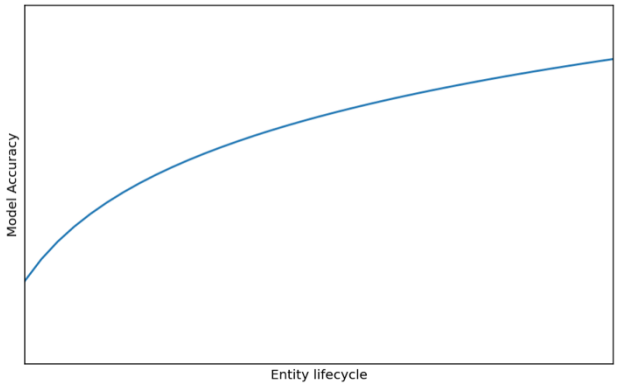


Fig.3 Summary representation of model accuracy of an entity over entity life cycle

Since the modelling accuracy has a dependency on the training data, we can extrapolate a similar relationship to exist between the model accuracy metrics and the chronological point in an entity life cycle.

Models that are trained on data in a chronological instance will generally have a higher accuracy that a similar model (assuming the same Machine Learning algorithm with the same set of hyperparameters) trained at a previous instance. Thus, the relationship between the accuracy of the modelling metric across the entity lifecycle can be represented graphically as below in figure 3.

As discussed in the earlier sections of this paper, the only way to derive value from Machine Learning developments is to act or intervene on the insights generated by the algorithms trained on data.

Ideally, the ideal insight and thus the subsequent intervention should be performed at the beginning of the entity life cycle, in the hypothetical example the most effective intervention for addressing customer churn could be to not proceed with customers who would churn based on just the self-entered information as a prospect / lead and the least effective intervention would be at stage just before the churn event as that will lead to no business value.

Thus, the relationship between intervention from insight can and the entity lifecycle is of an inverse nature and be represented graphically in figure 4.

Combining the relationships across data quality and the relevant intervention across the entity life cycle, we can observe the patterns to merge with a brief overlap as represented in figure 5.

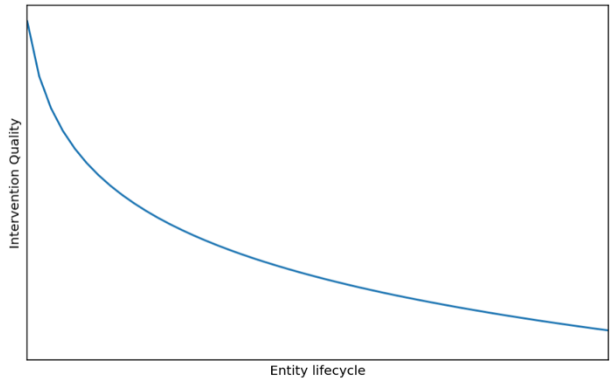


Fig.4 Summary representation of business intervention quality / effectiveness over entity life cycle

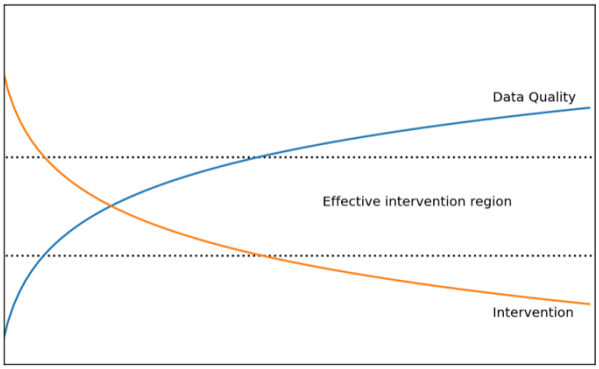


Fig.5 Representation of an inverse relationship between data completeness and business intervention effectiveness over entity life cycle along with an optimal region for intervention

Any effective intervention to realize business value must thus be undertaken within the bounds which can be described as the effective intervention region.

The most effective intervention region is spread across a duration of the entity lifecycle and the ideal point of intervention would be a tradeoff between the value and the richness of data at that point of the entity's lifecycle.

IV. Illustrative Example of a ML Based BPR

In the previous section we considered entity lifecycles and the subsequent metrics around data richness, model accuracy and the business value from interventions on insights. We shall consider the example of a credit lifecycle across a fixed duration lending financial credit to elaborate the maximization of value from using an ML based BPR approach.

Consider a generic customer life cycle for a fixed duration lending. The business objective is to minimize losses and write-offs across customers to whom credit was provided. Thus, the metric to improve would be portfolio loss rate. The current metric can be defined as the Base Loss Rate (BR), the metric post intervention shall be called as the Current Loss Rate (CR).

The various steps in this process could be represented along with the data availability at that stage as shown in figure 6. A customer can drop out at any stage / be rejected at any stage.

The most effective scenario would be to reject defaulting customers in the Lead & Credit check stage as there would be 0 loss in these 2 stages and the difference between the BR & CR would be the maximum at this stage. Since credit checks require an operational cost, the ideal scenario would be to reject the defaulting potential customers in the lead stage only and consequently the least effective intervention point shall be to predict and act on the customers towards the end of their repayment tenure, the difference between BR and CR would be the lowest at this point which aligns with our graphical representation of intervention effectiveness / value as discussed earlier.

In this scenario the most optimum intervention would be during a stage which provides the lowest loss and the lowest operational expense while ensuring good confidence in our predictions. It is important to note that the loss rate could be made 0 theoretically by rejecting all customers and thus the

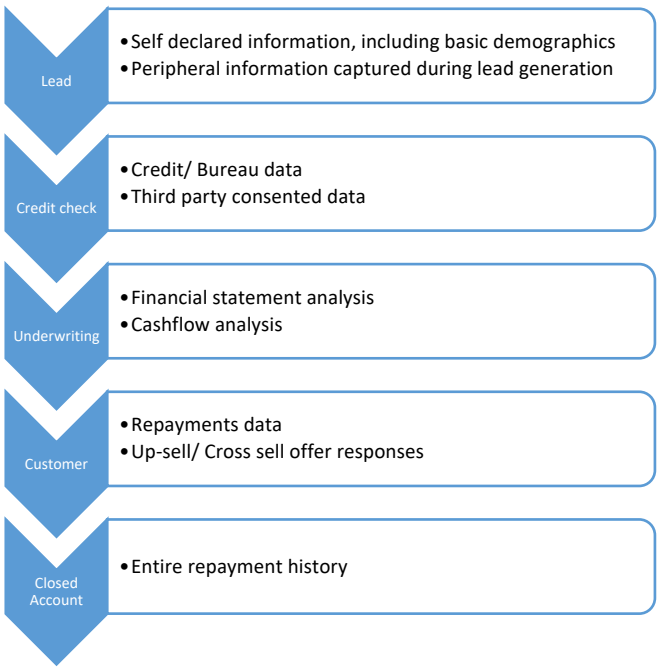


Fig.6 Chronologically generic steps of a financial lending product

constraints must be accurately defined before a trade-off is made.

In the current scenario, we can see that the stage that provides the most confidence in the predictions is at the underwriting stage when credit and cash flow data is available. However, at this stage 2 different interventions can be made, which are to completely reject the customer or to group high risk customers and lend to them at a high interest rate to offset the losses from this group. This however is not in the scope of the current paper as the value realization in practice would be spread across multiple risk levels and would involve portfolio optimization as it is practiced across the Banking and Financial Services Industry (BFSI).

This intervention can be tried for a smaller sample as a POC and the value of the process would be monitored carefully over an observation period, if the improvement in the primary metric of loss rates is significantly different then the same can be extrapolated to the population and the POC can be scaled up to a full production roll out.

V. Recommended Checklist for a Machine Learning Based Business Process Re-Engineering

Concluding the summary of the observations and insights shared earlier, the following checklist is recommended before working on development of a POC for an ML implementation

(1) A single metric to optimize and an established pre & post intervention calculated logic. If an exact calculation cannot be found or arrived at, it is not recommended to proceed ahead
(2) Identification of the entity life cycle stages and the corresponding addition of data points required. If discrete stages cannot be identified, then it is recommended that the continuous entity data entries are made discrete with fixed intervals
(3) Identification of the optimal points with strong confidence in the predictive power of the models

(demonstrated across out of time validations) and a positive ROI of the BPR development

(4) POC sampling, a smaller POC to be carried out on a smaller section / noncritical section of the intended function

(5) An observation window for the POC to be able to demonstrate a positive ROI in terms of value

(6) A scaled analysis of POC to full scope of deployment

(7) Roll back mechanisms to quickly restore the earlier process, roll backs are required as a risk management measure due to unforeseen circumstances which may arise due to either an ML based issue such as data or concept drift or be a business / regulatory / unforeseen event

VI. CONCLUSION

Machine Learning is one of the most promising technologies which is on the cusp of altering civilization. This has been further accelerated by the increasing penetration of digital transformation. Effective implementations of Machine Learning implementations, conversely, require carefully deliberated understanding of the process where an intelligent insight can be generated from an acceptable quality of data.

Data acquisition and quality improves over an entity lifecycle and inversely Machine Learning insights decreases over the entity lifecycle. Thus, an optimal intervention period for Machine Learning implementations must be analyzed and leveraged for the most effective implementations that drive the maximum impact.

REFERENCES

[1] Martens, Henrik H. "Two notes on machine “Learning”." *Information and Control* 2, no. 4 (1959): 364-379.

[2] 김재경, and 성태경. "Application of Visual Decision Making Process in the Development of Business Process Reengineering Vision and Implementation Plan." *Journal of the Korean Operations Research and Management Science Society* 14.2 (1989): 185-185.

[3] Williams, Henry. "Business process reengineering (BPR)." *Health Executive* (1990): 36-9.

[4] Hammer, M., and J. Champy. "Introduction to business process reengineering." *Industry week* 1 (1990).

[5] Lawrence, David., Solomon, Arlene. Managing a Consumer Lending Business. United States: Solomon Lawrence Partners, 2002.

ADaSci | THE ASSOCIATION OF DATA SCIENTISTS

BANGALORE, INDIA

ONLINE CONTENTS AVAILABLE: EVERY QUARTER ON **www.adasci.org/journals**