National College of Ireland

Elsagate Content Classification in Cartoon Videos

MSc Research Project
Data Analytics

Bhagyashree Sanjay Tatiya
Student ID: 18188788

School of Computing
National College of Ireland

Supervisor: Mr Manaz Kaleel

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Bhagyashree Sanjay Tatiya
Student ID:	X18188788
Programme:	MSc In Data Analytics **Year:** 2019-2020
Module:	Research Project
Supervisor:	Mr. Manaz Kaleel
Submission Due Date:	17th August 2020
Project Title:	Elsagate Content Classification in Cartoon Videos
Word Count:	7327 Words

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: .....

Date: 17th August 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Elsagate Content Classification in Cartoon Videos

Bhagyashree Sanjay Tatiya

x18188788

Abstract

The tremendous rise in digitization has led to various changes in the daily lives of every individual. Most of the people as well children are very prone to the use of mobile phones now and then. Many kids often watch cartoon either on Television or Mobile phones. Some cartoon videos might contain disturbing scene such as Pornography, Violence which are not appropriate for them to watch. To solve this problem various websites as well applications are launched which were only made for children to watch the cartoon. Although the problem was not solved completely, these scenes were restricted to a great extent. So various techniques have been implemented to solve this problem. The use of Deep Learning Techniques such as VGG-19, Bi-LSTM, Autoencoder is done here to classify the videos into four different categories such as the Sexual, Violent, Both Sexual and Violent and None. This classification of the videos would eventually lead kids to watch cartoon videos without parental support.

Keywords: Deep Learning, VGG-19, Bi-LSTM, Autoencoder

1 Introduction

Nowadays every individual is surrounded by technology. Technology has entered every aspect of society. Today's kids are intelligent enough to use a mobile phone or television. They are capable to navigate themselves to their favourite game or video playing platform. Consumption of videos has increased over the past few years, wherein these videos can be used for education as well as entertainment. Kids mostly use them for entertainment such as watching cartoon videos (Singh et. al, 2019). The platform such as YouTube Kids, Hopster, Kidoodle etc. have been launched so that kids would watch a cartoon without parental supervision (Tahir et. al, 2019).

Cartoons have become one of the most important sources of entertainment for kids in recent times. The impact of cartoons on kids has increased in the last decade. Children watching cartoon start depicting those acts in their day to day behaviour. These cartoons videos could be used for the development of children but at the same time, these videos can impact the kid in a negative way. Many a time there are vulnerable acts which are not good for kids to watch. These types of acts impact the kids mentally. They try to imitate these acts in their behaviour which not only affects them but also society. According to a research, it is proved that children who are used to watch such violent cartoons on regular basis are more inclined towards performing such activities on daily basis (Khan et. al, 2019).

YouTube has become one of the most important sources of entertainment as well as for children and adults. YouTube has a plethora of videos in various fields. There are some videos which contain scenes such as violence, sexual scene, pornography etc. These types of scenes are not appropriate for children to watch. So, to provide a safer environment to children YouTube has launched YouTube Kids. Apart from this, various other children safer websites and applications such as Kidoodle, Hopster, KidsCast etc have been launched. Despite so many websites and application, there is some child unsafe content available in cartoon videos such as violence. For example, YouTube relies on user's perspective on that video. If a user reports a video to be unsafe then YouTube manually checks that video for unsafe content (Papadamou et. al, 2019). However, as the whole process is manual it is very difficult to inspect every video, as YouTube serves a large number of videos.

To automate the process of inspecting the videos for unsafe content Deep Neural Network comes into the picture. This Network will help in detecting inappropriate content for children. Two types of classes are studied here, and they are violent and sexual. Combining all, we have four different classes namely violent, sexual, a combination of these two classes i.e. both and remaining all the scenes are considered as safe. In the previous research, frame-level approaches have been used to solve the problem where there was a need for lots of manual efforts (Ishikawa et. al, 2019). So, to automate the process Recurrent Neural Networks along with Convolutional Neural Networks are being used here to solve the problem. Also, the generation of frames from the videos is an automative process. Bidirectional Long Short-Term Memory (Bi-LSTM) unit have been used here for capturing the sequence of the frames being generated. The use of Bi-LSTM for video detection was proposed by Tahir et. al, 2019. VGG-19 is built, and features of every frame are extracted before Bi-LSTM for obtaining the video descriptors that are being generated by it. Bi-LSTM along with Autoencoders are used to work with the size of the video clip.

The organization of the report is such that Section 2 contains all the Related Work i.e. the Literature Review of all the previous work that has been done the Classification of Images and Videos and the techniques used for such classification. The next section i.e. Section 3 of the paper comprises of the Methodology employed during the implementation phase of the system. Section 4 contains information about the Design specification of the System. The models used during the implementation phase. The next section gives information about the Development Environment of the system i.e. the software used, and the libraries installed. Section 6 is the Evaluation section which gives the result generated by the system developed. The next section gives the Conclusion and the Future Work of the system.

Research Question:

How well Deep Learning models can perform in Classification of Elsagate Content (Sexual Scene, Violence) in Cartoon Videos?

2 Related Work

The primary source of entertainment for today's kids is Mobile Phones. Children are addicted to it to a very great extent. Kids use a mobile phone to watch cartoons on YouTube or any other video streaming application or to play games. The cartoon videos which are being streamed by these applications many times contain scenes which are not appropriate for children to watch. Different scenes such as violent acts, sexual content are inappropriate for children to watch. So, to classify these videos into different categories Deep Learning Techniques are used (Papadamou et. al, 2019, Singh et. al, 2019). Deep Learning is a branch of Computer Science which has various techniques for the classification of videos. Machine Learning also has a significant amount of techniques that could be used but Deep Learning has paced the training and evaluation part of the techniques to be used. The use of Deep Convolutional Neural Networks was done by Ishikawa et al. (2019) where the cartoon videos were used to generate frames. These frames were further passed through the Convolutional Neural Network (CNN) which helped in the classification of videos. Different mobile architectures were used along with CNN such as MobileNet, NASNet, SqueezeNet etc. has given significant contribution using Transfer Learning.

A 3D-CNN network is developed using different optimization function. The videos which are used here for classification are firstly converted to frames which are passed through the different layers of the CNN model. The 3 optimizers namely Stochastic Gradient Descent (SGD), Adagrad Optimizer and Adam Optimizer are used here to check the accuracy of the model. The Accuracy obtained by Adam Optimizer is highest which is 96.1% whereas the accuracy obtained by the other two optimizers is slightly less as compared to Adam Optimizer. (Jiang et. al., 2019)

2.1 CNN, LSTM, Autoencoder

The instalment of surveillance cameras in various public places has become mandatory because of the increasing crime rate. The manual monitoring of these cameras is a tedious task. So, Das et. al. (2019) has used Machine Learning techniques for the monitoring violence in the acts. The videos of Hockey game were used wherein frames are generated from these videos. Features were extracted from these videos. Apart from this, the Histogram of Oriented Gradient (HOG) was used to extract low-level features from the frames. The system is trained using six different Classifiers namely Support Vector Machine (SVM), Logistic Regression, Random Forest, Linear Discriminant Analysis (LDA), Naïve Bayes, k-Nearest Neighbour (k-NN). The use of different classifiers was done here to see how the result generated from these classifies as the features are extracted using HOG. Evaluation Techniques such as Confusion Matrix, Accuracy, Recall, F1-score, Precision were used. Highest accuracy of 97% was found in the Training phase of k-NN which seems to be exactly opposite in the testing phase. In the testing phase, its accuracy is as low as 81.5% and Precision is also very low. Random Forest maintains a good score in both Training and Testing Phase. Also, the detection of violence in CCTV cameras using Deep Learning Techniques have performed better. Perez et. al., (2019) proposed the use of CNN along with Support Vector Machine (SVM) and Long Short-Term Memory (LSTM) was done. The

author has used feature detector and feature descriptor called Temporal Robust Features which is used to extract Spatio-temporal features from the frames. The model predicts the type of frame i.e. whether the negative or positive case depending upon the training provided to the model. After all the predictions, the aggregation of all the result generated in the previous step is done to predict the fight instances. This method uses 2D-CNN architecture. The Mean Average Precision(mAP) obtained in Two Stream CNN is 79.5% whereas the F-measure for LSTM is 75.9% which is much higher than the remaining methods.

Image classification has become one of the important progressions in the field of Deep Learning. Here, Lai et. al (2019) has used the images of the cloud for classification. Around 2300 images of eight types of cloud are used here. These images are fed into the network of VGGNet wherein different layers of the network process these images by extracting the features from the images. The inception networks used here along with the VGG16 has helped in decreasing the error rate of the model. As the images of clouds get classified, in the same way, images of the flower also get classified. The dataset used by Cengil et. al (2019) contained 5 classes of flower each having 800 images. The use of Transfer Learning is done on different networks such as AlexNet, ResNet, VGG16, GoogleNet, DenseNet. All these networks gave different result according to their time required for processing. The time required by DenseNet and VGG16 is almost nearby and the accuracy obtained which is 93.06% and 93.52%. The accuracy obtained by AlexNet, ResNet and GoogleNet does not show a major difference between them which is 86.28%, 91.29% and 89.75%. The time required for validation is more or less the same. The highest accuracy is obtained from VGG16 as it is built up using the important aspects of the CNN model. The deep construction of the model increases the accuracy of the classification done.

Emotions to be expressed by the medium of the face is an integral part of human activity. So, the proper classification of images into different classes depending upon the facial expression is important. The Audio-Visual dataset of Emotional Speech and Song was used here to classify the videos into different categories (Abdullah et. al, 2020). Use of CNN along with LSTM id done here. Frames are generated from the videos which are then passed through the network of CNN and LSTM. Xception Net was used to extract frames from the videos. The validation accuracy obtained from these videos is 80%. The model which was trained using these videos was able to achieve a test accuracy of 61%. Apart from classifying human expression, Deep Learning approaches can also be used for classification of sports videos. A small dataset which has 5 different variety of Sports is used. Frames were extracted from these videos which were then passed through the combination of CNN as well as RNN i.e. Gated Recurrent Unit (GRU) model which has achieved an accuracy of 96.66%. Another model which contained less parameter than the previous model which has a test accuracy of 80%. A model built using different algorithm namely VGG-16 and Transfer Learning was used which has achieved test accuracy of 94% and training accuracy of 100% which outperforms the previous models (Russo et. al, 2019).

Rather than capturing videos to recognize human, sensors used in smartphones can also be used for activity recognition (Deep et. al, 2019). Earlier, Machine Learning techniques were used but the advancement in technology led to the use of Deep Learning Technique which would enable more fast and accurate results. Human activities are captured through the sensors which are present in the smartphone. This data is further passed through the CNN and

LSTM networks. The data which is passed through the CNN layers are converted to a 4D matrix. Further, this matrix is passed through the LSTM layers to predict the activities. Here, the dataset comprising of 6 different activities is used. As LSTM is capable to capture the temporal dependencies from the data. The result showed the accuracy of 92.98% by using only the LSTM model, whereas the accuracy obtained by using CNN and LSTM is 93.40%.

Motor imagery classification is an important task. Deep learning has gained attention in many areas and has been widely utilized in all those areas for performing a various task. Brain-Computer Interaction (BCI) is a field which has lacked the use of Deep Learning (Lu et. al, 2019). The use of CNN along with LSTM is done here for classification of images. The output of the CNN model is fed to the LSTM network. As the number of LSTM unit increase the accuracy obtained from the network also increases. According to Ahmad et. al (2019), the number of CNN in the network can also be increased to gain more accuracy. The CNN used in this network is plotted parallel to one another wherein the output of every CNN network is merged and passed through the LSTM network after which it passes through the Fully connected layer. This is the last layer where the final classification takes place. The data used here is from the sensors which capture the human activity which comprises of 6 different activities. The author has made use of Machine Learning models such as k-NN and SVM which gave a good precision value. But the accuracy obtained from the Single Head CNN was much higher than these algorithms which are 94.1%. Also, the Accuracy obtained from multi-head CNN is 95.76% which overtakes Single Head CNN.

An anomaly detection system built using the CNN and the autoencoder for detecting anomalous acts from the videos. A spatiotemporal autoencoder along with the CNN is used here. Apart from this Representation Learning is used which can help predictors and classifiers to extract useful information from the images (Nayak et. al, 2020). The whole process comprises of Spatial Encoder, Temporal Encoder-Decoder and Spatial Decoder which are made up of CNN and LSTM model. The regularity score generated from the frames is used to decide whether the frame is anomalous or not.

2.2 VGG16, Bi-LSTM

Nowadays, Automated Teller Machine (ATM) is the necessity where people use it to withdraw cash. As the use of ATMs has increased, thefts related to it has also increased. Although there are surveillance cameras are been installed the theft has not decreased much. So, a Deep Learning approach is been used here to decrease the theft related to it (Parab et. al, 2020). CNN along with Bidirectional LSTM (Bi-LSTM) so that the person performing such anomalous activities to be caught red-handed. The cameras which are installed near the ATM were used to capture the videos which are then passed through the CNN where the feature extraction would take place. These features are then passed through the Bi-LTSM which has the capability of remembering the features which can be further used for classification. This system gains an accuracy of 82% by classifying the videos in anomalous and non-anomalous activities. As soon as the activity is classified into anomalous act the nearby security authority is informed and the person is caught. The whole process takes place in real-time. Apart from installing cameras near ATM, they are also installed in shopping malls, hospitals etc. These cameras record Human activities to analyse the anomalous behaviour in it. The videos recorded in these cameras are used to generate frames which are

then passed through the CNN network which does the process of feature extraction. As the number of frames generated from the videos is large, ResNet is used to increase the speed of training and learning (Mihanpour et. al, 2020). The frames are generated from the videos which mean there would be dependency among frames. As a result, a model which is capable to remember the previous and the next frame is used, namely the Deep Bi-LSTM model. This helps in detecting the complex and sequential pattern in the frames. The proposed method gives an accuracy of 94.79%.

The use of VGG-16 in the classification of Transmission Line was done by Zhang et. al (2019). By replacing a combination of Convolution Layer with the Fully Connected Layer which can improve the extraction of features from the images. The use of Optimized Convolutional Neural Network was done here. The Optimized VGG-16 gave the validation accuracy of 95.1% which was much higher than that of the VGG-16 which was 88.7%. Optimized VGG-16 is made up of Convolutional layers along with the ReLU. Apart from this, there is another layer which is made up of a combination of Max Polling Layer, a Sigmoid Function with the Convolutional Layer and Flattening Layer. The final output obtained from this is the Classification result of the Transmission Lines. Liu et. al (2019) has also use of Machine Learning Algorithm termed them as Shallow Machine Learning Algorithm such as Decision Tree, Logistic Regression, SVM, Random Forest. Every algorithm gave different accuracy in a different scenario. Accuracy of the model varies ranging from 59.51% to 75.15% whereas the different Deep Learning models such as the VGGNet, RS-VGGNet, Deep Sat Network, Patch-based CNN. The accuracy obtained from the RS-VGGNet is 99.6% which is the highest among all. Tian et. al (2019) has proposed a model for the personalized design of men's clothing. The use of Deep Neural Network is done here so that men's shirt could be according to their choice. All the data is passed through the Bi-LSTM network. This data is checked with the Clothing model Database which generates a pattern for the shirt according to the user's choice. If the customer satisfied, the order is generated and if not then the restructuring of the order is done. This method generates virtual clothing which allows the customer to decide whether to make changes or not.

VGG-11 was used here for the detection of Object Forgery present in the videos (Yanfen et. al, 2019). The author has proposed a model using Deep Learning Technologies based on VGG-11. The video frames when first passed through the model, it calculates the motion residual map of each frame is created while the extraction of Steganographic features. Before passing these Steganographic features of the motion residuals, the fully connected layers is implemented to transform the dimension of the features to a specified size. The Activation function used here is ReLU. After the Fully Connected Layer there comes a Convolutional Block followed by the Pooling Layer. And again, at the last stage of the model, there is a Fully Connected Layer through which the final output is obtained. The accuracy achieved by the Classification of the Frame is 98.19%. Supervised Learning Model has been implemented on the Time Series Data for Binary Classification (Zhou et. al, 2019). This result obtained by the classification of data is evaluated by using the Confusion Matrix. The Classical point-to-point Confusion Matrix gave an improper result for the model performance. So, a new approach towards Confusion Matrix for evaluation of imbalanced

time series data is implemented. Also, the ROC curve is plotted for the classification result obtained from the binary Classification.

3 Research Methodology

For any research and development of a project, there should be a methodology that needs to be followed. There are many methodologies which can be followed for a project such as Knowledge Discovery in Database (KDD), Cross-Industry Standard Process for Data Mining (CRISP-DM) and Sample Explore Modify Model and Assess. The objective of the project is to Classify the Videos into different Categories depending on the content of the video. This is achieved using various Deep Learning model such as VGG-19, Autoencoder, Bi-LSTM. So, KDD framework has been the best choice for the methodology that can be followed. Following the CRISP-DM framework can also be a better choice but the last stage in it is Deployment which cannot be achieved, so following the KDD framework is a better option chosen. The last stage of KDD framework is evaluation. Moreover, there are changes made to different phases of the framework according to the need of the research. The following Fig 1 illustrates the various stages of the KDD framework.

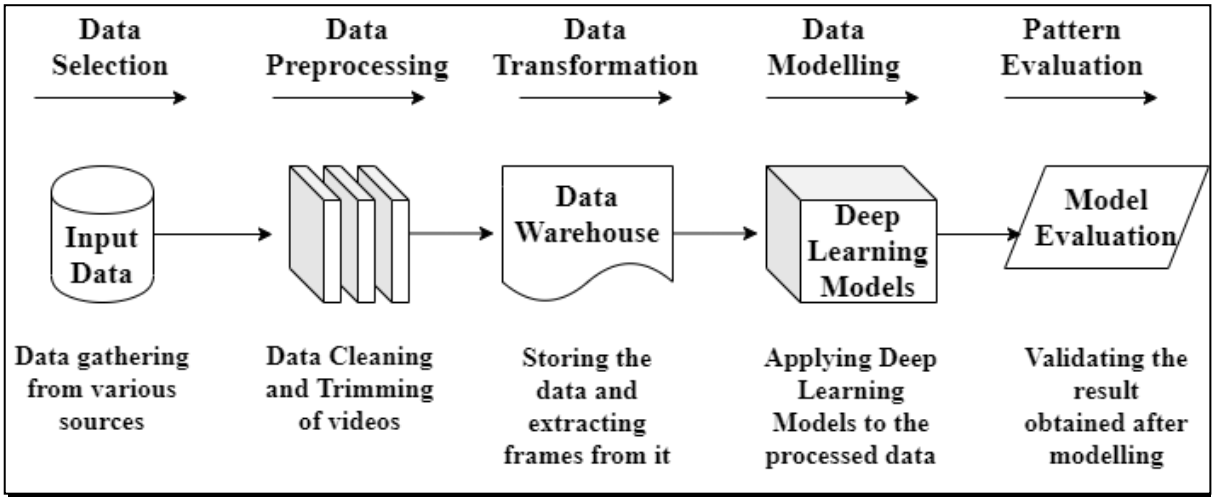

Fig 1: Process Flow of KDD

Data Selection

This is the first stage of the framework where the data required for processing and training of the model is gathered. All the videos required for classification are collected from various websites as well as application. Different types of cartoon videos are collected such as the videos containing the Elsagate content and normal funny content in cartoon video. Taking into account the time and computing resources, the dataset used for classification is limited to 40 videos. All these videos are downloaded and stored in the database for further processing.

Data Pre-processing

This is the second stage of the framework where the stored videos are trimmed into shorter videos of small duration. The videos which are stored are of 20-25 minutes which are

then trimmed down to smaller videos of 3-4 minutes each. There are various sections in the videos which are not required for processing. The trimming of the videos is done manually as the unwanted content is distributed throughout the video. Apart from this, converting them to short videos will also help in reducing the processing time and time required for the training of the models.

Data Transformation

This is a stage where all the videos which are trimmed and stored onto the database are used to extract frames from it. The extracted frames from these videos depict the whole story of the videos sequentially. The library used here is cv2 which has a function that can be used for extracting frames.

- VideoCapture(): Used for capturing the video from the given location.
- imwrite(): Used in writing the frame to the specified path.
- read(): Used for checking the end of the function. It is a Boolean function wherein if it returns true then the frame is generated and written to the path.
- set(): Used in capturing the frame at the specified time which is passed in this function in milliseconds.

Data Modelling

After all, the transformation and pre-processing of the data is done, there comes a stage where the Deep Learning model should be implemented through which the data is passed. These models help in the classification of videos into specified categories. Deep Learning model VGG-19 is implemented first which is a Convolutional Neural Network model mainly used for Image Processing. This model has 16 Convolutional Layers where every layer is followed by the ReLU activation function and some of the Convolutional Layers are followed by Max Pooling Layer. Also, it has 3 Fully connected layers which bring the total of Convolutional layers to 19. The data produced by this model is passed through the next model called as the Bidirectional Long-Short Term Memory (Bi-LSTM). Bi-LSTM helps in improving the model performance on Sequential Classification problem. It trains 2 LSTM structure instead of one LSTM to remember the past and the future of the data. The output of this model is further passed through the Autoencoder. The Autoencoder and Bi-LSTM work together in this process. Autoencoder helps in reducing the size of the data obtained from the VGG-19. The final results are obtained at this stage. The output of this stage is further passed for evaluation where the accuracy of the results is checked. All the above models are implemented using the torch library which has a various function which helped in building the model.

Model Evaluation

The application of the model onto the data is a step where the classification of data is done. To evaluate the accuracy obtained from the model different evaluation techniques are been used. As the problem here is the classification problem then Confusion Matrix is best suited here. Use of Precision and Recall can also be done here for checking the accuracy of the model. A description explaining all these metrics is done below.

- Confusion Matrix: A Confusion Matrix is a table-like structure which is used to report the performance of the Classification that is being done on the test data. It gives the visual presentation of the performance of the model. It gives us an idea about the types of errors being made by the model. True Positive, True Negative, False Positive and False Negative together make the confusion matrix.

- Accuracy: Checking the Accuracy of the result obtained, helps in evaluating the efficiency of the model where there is a classification problem. Also, it gives the number of accurate predictions made by the model over the total data. It indicates the percentage of accuracy obtained after the classification done by the model.

- Precision: Finding the precision metric of the model involves considering various measures. It is the ratio of several cases where the model accurately predicts the class of test data divided by the sum of accurate predictions done by the model over the test data and the wrong predictions done on the right data.

$$Precision = True\ Positives / (True\ Positives + False\ Positives)$$

- Recall: Recall is another evaluation metric which calculates the percentage of accurately classified classes divided by the sum of accurate classification and incorrect classification done by the model.

$$Recall = True\ Positives / (True\ Positives + False\ Negatives)$$

4 Design Specification

The research project here makes use of 2 tier architecture as shown in Fig 2. The first tier is the Business Logic Tier and the second tier is the Client Tier. There is no need of the third tier as the dataset was downloaded from different websites, instead of using any API or cloud-based approach for accessing the data. Thus, eliminating the need for the third tier. The main tool for the implementation of the model used here is Python as it has various libraries such as Torch, CV2 etc which can be easily imported to use all the functionalities required and accordingly the models would be developed. Jupyter Notebook served as an Integrated Development Environment (IDE) where different libraries were imported for processing and transformation of data. These steps were further followed by the implementation of different models such as the VGG-19, Autoencoder, Bi-LSTM. The result obtained from all these models is evaluated and then visualized using the matplotlib library. Plotting of the output obtained from the models allows us to easily interpret and compare the result obtained. The below figure gives a pictorial view of the whole processing that takes place and the interaction taking place between them.

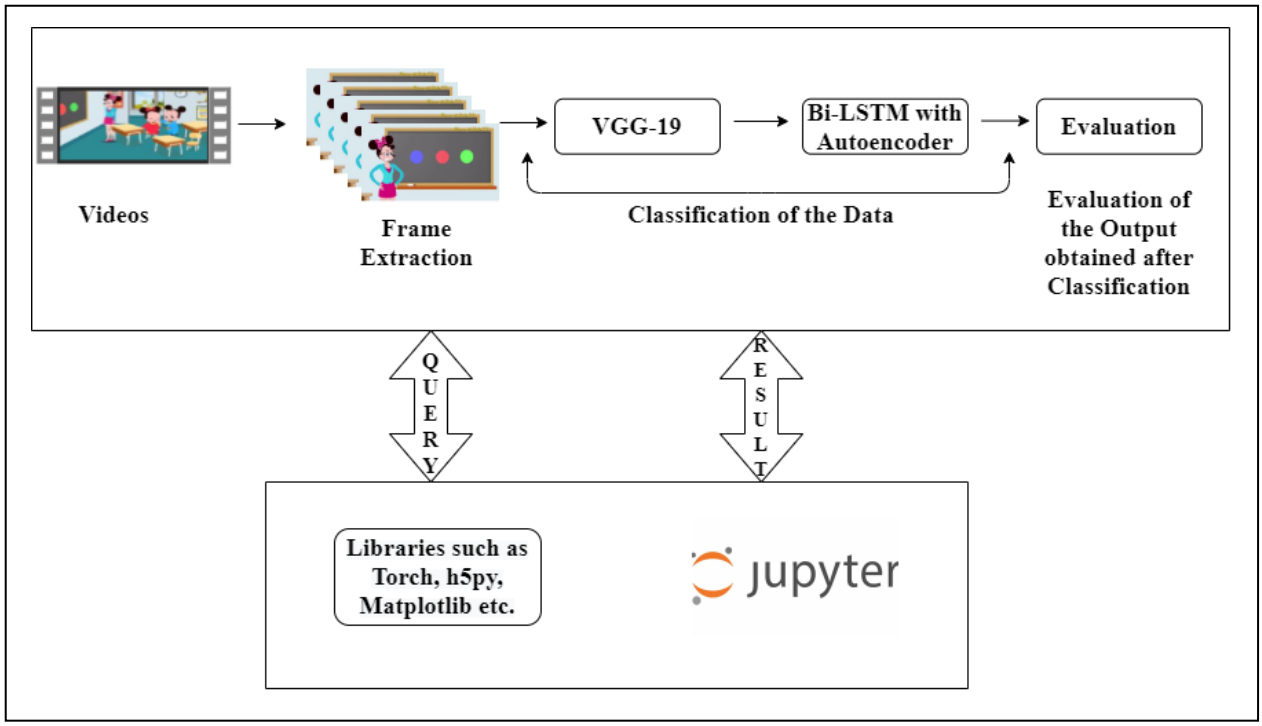

Fig 2: System Design

4.1 VGG-19

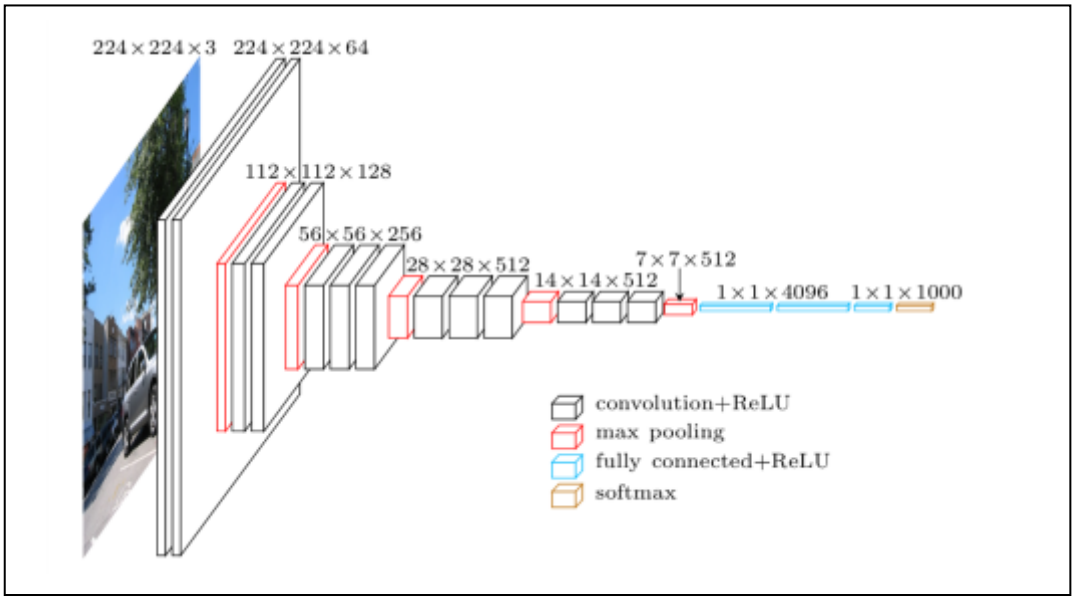

Fig 3: VGG-19 Architecture

VGG-19 belongs to the family of Convolutional Neural Network which also a part of much Deep Learning model used for image classification. Many Machine Learning models can improve their execution time by using the Feature Extraction Technique such as Histogram of Oriented Gradients (HOG), Scale Invariant Feature Extraction (SIFT). These techniques are complex and require a resource person for implementation. These techniques

have lots of importance in image processing but when it comes to real-world deployment of these machines, they may not be feasible enough to handle the large amount of data generated. These machines may not be compatible enough for identification and classification of the frames. The VGG-19 architecture used here does the process of feature extraction and segmentation of the image provided as an input, as there are layers in the model which helps in doing it. The functionality provided by these layers of the model proves to be more effective than other models in a similar area of study. The Fully Connected Layer helps in classifying images where each neuron is connected to all the learned feature maps. The functionalities of the layers present in the model are described in Fig 3.

- Convolutional Layer: This layer is used to extract features from the image while storing then into a different file. While extracting these feature they make use of the filters that learn from the input data i.e. image. It receives the input as the extracted frames of the videos that are convolved with the filter to extract the basic characteristics of the image such as curves, edges, a dimension of the object and generates a feature map.
- ReLU Layer: The activation function called a Rectified Linear Unit (ReLU) which changes all the negative activations to 0. The feature map generated above is passed through this where all the negative values are changed to 0. This layer does the non-linear transformation of the data without affecting the receptive fields of the Convolutional layer.
- Pooling Layer: This layer reduces the dimensions of the images which helps in reducing the computational power required by the remaining layers of the model. This layer is used in the model which selects the maximum value of the region based on the filter size.
- Fully Connected Layer: These are the part of Convolutional layers of VGG-19 which are capable of identification of features that are correlated with the output class. The last 3 layers contribute to the Fully Connected Layer of the 19 Convolutional layers. The output obtained from this is a one-dimensional vector which is obtained by the flattening process done on the results obtained by the previous layer.

4.2 Bi-LSTM

Bi-LSTM is Bidirectional Long Short-Term Memory which is the advanced version of LSTM. Bi-LSTM has the capability of improving the model performance when there is a Sequence Classification Problem. When the input sequence along with their time steps, then the Bi-LSTM trains two LSTM on the input sequence. The first sequence which is the Forward Layer is trained using the same input sequence but the second sequence that is the Backward Layer is trained using the reversed copy of the input sequence. It means that this sequence of Bi-LSTM is capable of remembering the past as well as the future of the input sequence.

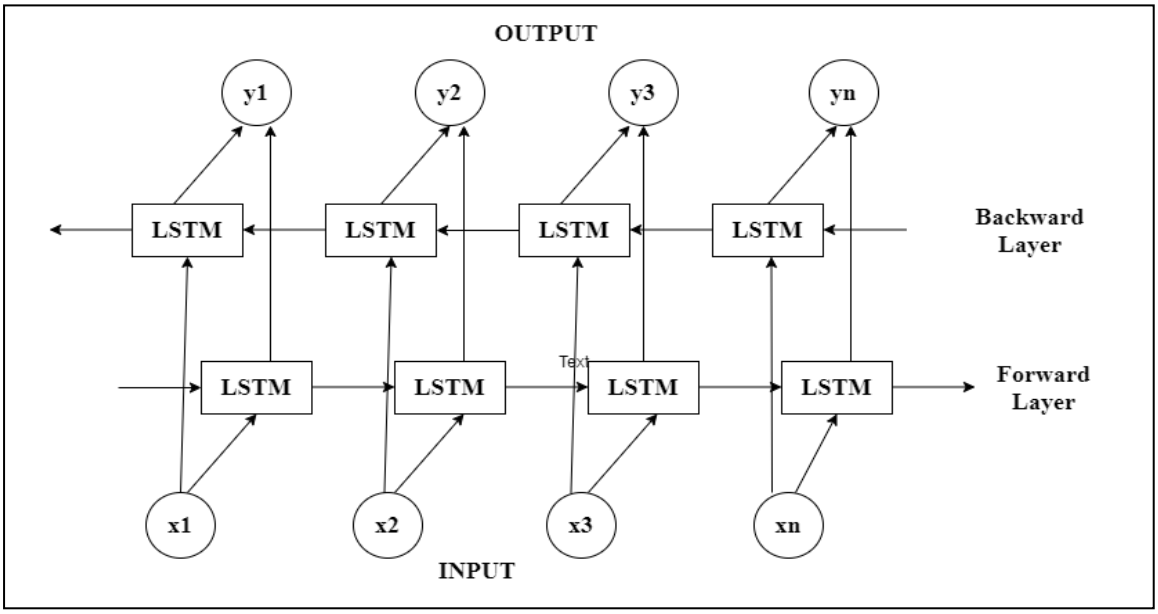

Fig 4: Bidirectional Long Short-Term Memory

4.3 Autoencoder

Autoencoders are a type of feed-forward neural network that can represent the data by reducing the dimensionality and noise in the network. It can efficiently compress the data by encoding it and then again reconstructing the data from the encoded data to represent data that is almost similar to the original data. Below is Fig 4 which shows the structure of Autoencoder. The input nodes are the nodes where the data is passed, and all the encoding of the data takes place. The hidden nodes are the nodes where the compression of the data takes place. There are many hidden nodes as well as many hidden layers. The output nodes are the nodes where all the decoding of the data is done.

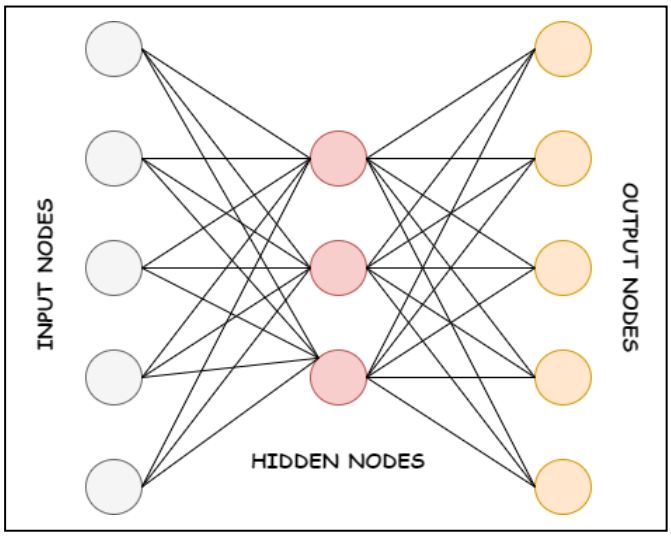

Fig 5: Autoencoder

Autoencoders are data specific means they can only compress the data on which they are trained. They learn the features of the data that is given to them for training. The output produced by the Autoencoders is lossy as the compression of data takes place here. The output data would be similar to the input data. While training the autoencoders, it is not mandatory to use the labelled data. They can label the data by themselves. They are considered to be the Unsupervised Learning Technique as they don't require labelled data.

The structure of Encoder, as well as the Decoder part of the architecture, is almost similar. The Decoder part is almost the similar reverse replica of the Encoder part. The Encoder part reduces the dimension of the data while performing compression in the Latent space *l.* If the input given to the Encoder *E* is denoted *x* then the expected output from the Decoder *D* is denoted as *y*. Then the output produced by the Encoder is given as $l=E(x)$, while the decoder accepts this output constructing the original input by minimising the loss and removing the noisy data. The Decoder *D* and the output obtained from the Decoder part is *o,* which is given as $o=D(l)$. As the number of layers increases the result obtained from the autoencoder becomes more precise. As the layers increase the number of nodes in each layer also increases. The number of layers allows the autoencoder to handle more complex data. Here, Autoencoder is implemented with the Bi-LSTM structure of the model.

5 Implementation

This section focuses on the details of the steps that are followed during the implementation of the classifier for the Classification of Elsagate content in Videos.

5.1 Development Environment

The most common programming languages used for the implementation of Machine Learning and Deep Learning models are R and Python. The research project here is purely implemented using Python with the Jupyter Notebook which is included in Anaconda and which serves as the Integrated Development Environment (IDE). Python was mainly chosen as a programming language because it has various libraries in it which can help in building a Deep Learning model. The library such as Keras, Tensorflow, Pytorch has made the implementation of Deep Learning models easier. Other packages which help in performing the basic functionalities such as NumPy, os, scikit, sklearn etc. are used here. Apart from all these matplotlib was used to plot the results generated from the model.

5.2 Data Handling

As discussed above, the dataset required for the implementation of the project was download from YouTube being the main source of data. Apart from this, various cartoon videos were also downloaded from various websites. As these videos were of longer duration trimming of these videos was done to have small videos which reduce the computation time of these videos. After this, annotations of every second of the video were done by classifying every second of the video into 4 different categories namely Violent, Sexual, Both Violent and Sexual and lastly the None category. These Annotations were written in a text file which was further used for classification.

A library called as cv2 was used here for extracting frames from the videos. Here, 3 frames were extracted from every single second of the folder and stored in different folders

named according to the second. After this annotation file was created which using the frames and the annotated text file generated before for further processing of data. Further sklearn library was used for partitioning the data into 2 different parts namely the test data and the train data.

5.3 Model Implementation

The models discussed in the above sections are implemented and the output generated from them is explained below

5.3.1 VGG-19

The above code snippet gives an idea of the implementation of the VGG-19 model which has the capability of extracting the features of the image in the further steps. The development of the model whose architecture is pre-trained can be downloaded using the library torchvision.models . The weights of the model which are pretrained are downloaded onto the machine running the algorithm. VGG-19 uses the current project for finding the feature matrix which will be passed further for classification. However, the model downloaded is huge and requires a larger space of around 500 Mb. The model was run using the frames as input data with the size of 640*360 pixels. After passing the extracted frames from the model, a multidimensional array of those features is created which are stored in one single file. The type of file used here for storing the data is HDF5 file which is created by using the python package called h5py. This package provides an interface to the binary data stored in the HDF5 file. This file has the capability of storing huge numerical data which can be easily manipulated using the python package NumPy. It has the capacity of storing thousands of datasets which can be categorized according to the user's choice. Extraction of features from the frame is a very time-consuming process and requires lots of patience. As the size of a tensor which is used here for storing the features of the image keeps on increasing, so whenever a new feature is added it needs to be resized.

```
vgg19 = models.vgg19(pretrained=True)
layers = list(vgg19.features.children())
layers.append(nn.AdaptiveMaxPool2d(1))
modified_vgg19 = nn.Sequential(*layers)
for p in modified_vgg19.parameters():
    p.requires_grad = False
```

Fig 6: VGG-19 Model Implementation

As discussed above, annotation of every second of the video is done which are in string format. Here, those annotated string are given numeric values such as 0 for none, 1 for violent, 2 for sexual, 3 for both. So, according to the string the numeric values are placed. All these annotated data are stored in a separate HDF5 file. After this, a separate HDF5 file is created where the aggregation of the features of all the frames and are stored together. Further, the aggregation of the clips is also done using the annotated data which is stored in a different file. This generated data is used to create data which is suitable enough to be passed through the further models for classification. After this, the data is divided into train and test.

The training data contribute to 80% of the data, while the testing data contributes to 20% of the data. This training and testing data is saved into a different file for further processing.

5.3.2 Bi-LSTM and Autoencoder

The implementation of Bi-LSTM and Autoencoder is done here. The Autoencoder used here does the process of encoding and decoding the module. All the encoding and the decoding of the model are saved in a separate file. A separate file for loading the training functions and separate file for loading the torch functionalities for creating the data is written in python and imported into the program. The data generated after the encoding and decoding is saved in a different tar file which is further used for Bi-LSTM where the final classification of the data takes place. The training time required for autoencoder is relatively less than the feature extraction process of VGG-19.

This data is further passed through the next model i.e. Bi-LSTM where the final classification of the data takes place. The loading all the data generated in the previous step is done after which training of the Bi-LSTM model is done here. The HDF5 file generated before is used here for training of the model. The use of torch.optim library is done here for optimization of various models that are implemented here. After the generation of results, they are stored in an HDF5 file which is further used for evaluation.

6 Evaluation

In this section, a detailed description of the analysis of the result obtained from the models for achieving the objectives of the project. The Deep Learning models that are implemented here which help in demonstrating the robustness and feasibility of such techniques in Classification of videos into different categories. Different measures are used here for evaluation of the result obtained from the models. Measures such as Confusion Matrix, Precision, Recall etc are used which gives the overall accuracy obtained from the model.

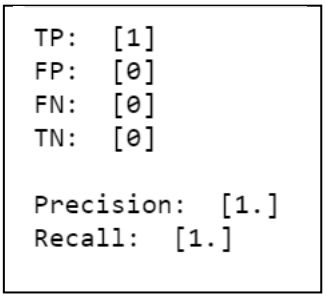

Fig 7: Output

The above Fig 7 gives the output obtained from the model execution. The values of Precision and Recall are mentioned in the figure. Apart from this the values which combine to form the Confusion Matrix are also shown. The four-measure called TP, FP, FN, TN are also shown in the figure.

6.1 Execution Time

Evaluating the models based on the results obtained is not sufficient. Evaluating the models based on their training time concerning the amount and type of data also matters a lot. In this research project, the amount of data used for training of the model is small, but the data is in the form of .mp4 file which are big files of around 200Mb-225Mb each. The process of extraction of frames takes around 1 hour as there are almost 40 videos of 3-4 minutes each and 3 frames are extracted from every single second by storing them into different folders. After extraction when these frames are passed further to VGG-19 model for feature extraction it requires almost 620 seconds for one epoch. Further when the annotations are being generated the time required is less. While passing the data through the Autoencoder the time required for the execution of one epoch is around 22-25 minutes. The execution time of Bi-LSTM is not every high and can be done easily. The final evaluation of the results is also very fast. This shows us that the system which is enabled with high-end GPU cores should be used for faster processing of the data.

6.2 Discussion

The motive of this research project was to build a model that would accurately classify the videos into different categories depending on the content present in those videos. A literature review was done to understand the current scenario of video classification. This mainly revealed that some amount of work was done in this direction as well. VGG-19 was used in the current research project. Along with this, different models were used to gain more accuracy in the result. Models such as Autoencoder and Bi-LSTM were developed to obtain more accurate results. Evaluation metrics such as Confusion matrix, Precision, Recall etc. were used for evaluation of the results. The pre-processing required for the data involving downloading of the videos, trimming the video, and preparing the annotation file for each video. Further converting of the images into tensor was done using NumPy. Different HDF5 files were also generated which has the output stored in the form of tensor i.e. multidimensional array. The file generated in one part of the code was passed to the next part of the code.

Although the research done was not successful enough in achieving its objective, it might be short of the data on which it is trained. Also, the research has faced many barriers, in the implementation phase it has performed significantly well. One of the most important barriers faced by the research is the amount of data used for training of the model and the computation time required by each model for training. Also, the data used for testing of the model was very less. This might be the cause of the model not performing well on unseen data. If the model would have been trained on a sufficient amount of data, then the result obtained from testing of the model would have been much better. Apart from all this, the very first step of extracting the feature of the frames is very time-consuming which can be improved by making use of High Configuration Machine. The amount of training time taken by these models was one of the reasons for training them on a small amount of data.

7 Conclusion and Future Work

The main goal of this research project is to classify cartoon videos into different categories such as Sexual, Violent, Both and None. This is done using Deep Learning Models such as VGG-19, Bi-LSTM and Autoencoders. These models have performed exceptionally well in Classifying the videos into different categories. Although the training time required by the model was much higher, the result obtained is significantly good enough. The feature extraction process done by VGG-19 is time-consuming but gave a promising result in the further steps of implementation. The various evaluation techniques are been used to see how well the model perform concerning the data and how well the classification of data is done.

The performance of Autoencoder along with Bi-LSTM can be improved in future. Also, Inception networks can be used in the implementation to improve the accuracy of the model. Also, Data Augmentation techniques can be used so that the result would be more precise. The model is trained using a single type of data so in future the variety of data can also be increased so that it can be deployed.

Acknowledgement

I would like to take this opportunity to thanks my mentor Mr Manaz Kaleel whose experience, guidance and encouragement have helped me a lot in completing my research project. I would also express my sincere thanks to all the faculty members of NCI who have shared their knowledge with us and made us capable enough in the field of Data Analytics. Last but not the least, I would also like to thanks, all my family members and friends for constantly supporting me and having faith in me without which I would not have been able to accomplish this.

References

[1] Abdullah, M., Ahmad, M. and Han, D. (2020). *Facial Expression Recognition in Videos: An CNN-LSTM based Model for Video Classification*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9051332 [Accessed 9 Aug. 2020].

[2] Ahmad, W., Kazmi, B.M. and Ali, H. (2019). *Human Activity Recognition using Multi-Head CNN followed by LSTM*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8994412 [Accessed 9 Aug. 2020].

[3] Cengıl, E. and Çinar, A. (2019). *Multiple Classification of Flower Images Using Transfer Learning*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8875953 [Accessed 9 Aug. 2020].

[4] Das, S., Sarker, A. and Mahmud, T. (2019). *Violence Detection from Videos using HOG Features*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9068754 [Accessed 9 Aug. 2020].

[5] Deep, S. and Zheng, X. (2019). *Hybrid Model Featuring CNN and LSTM Architecture for*

Human Activity Recognition on Smartphone Sensor Data. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9029136 [Accessed 9 Aug. 2020].

[6] Gan, Y., Yang, J. and Lai, W. (2019). *Video Object Forgery Detection Algorithm Based on VGG-11 Convolutional Neural Network*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9051239 [Accessed 9 Aug. 2020].

[7] Ishikawa, A., Bollis, E. and Avila, S. (2019). *Combating the Elsagate Phenomenon: Deep Learning Architectures for Disturbing Cartoons*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8739202 [Accessed 4 Apr. 2020].

[8] Jiang, B., Xu, F., Tu, W. and Yang, C. (2019). *Channel-wise Attention in 3D Convolutional Networks for Violence Detection*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8858306 [Accessed 9 Aug. 2020].

[9] Khan, M., Tahir, M.A. and Ahmed, Z. (2018). *Detection of Violent Content in Cartoon Videos Using Multimedia Content Detection Techniques*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8595563

[10] Lai, C., Liu, T., Mei, R., Wang, H. and Hu, S. (2019). *The Cloud Images Classification Based on Convolutional Neural Network*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9026121 [Accessed 9 Aug. 2020].

[11] Liu, X., Chi, M., Zhang, Y. and Qin, Y. (2018). *Classifying High Resolution Remote Sensing Images by Fine-Tuned VGG Deep Networks*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8518078 [Accessed 9 Aug. 2020].

[12] Lu, P., Gao, N., Lu, Z., Yang, J., Bai, O. and Li, Q. (2019). *Combined CNN and LSTM for Motor Imagery Classification*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8965653 [Accessed 9 Aug. 2020].

[13] Mihanpour, A., Rashti, M.J. and Alavi, S.E. (2020). *Human Action Recognition in Video Using DB-LSTM and ResNet*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9122304 [Accessed 9 Aug. 2020].

[14] Nayak, R., Pati, U.C. and Kumar Das, S. (2020). *Video Anomaly Detection using Convolutional Spatiotemporal Autoencoder*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9077128 [Accessed 9 Aug. 2020].

[15] Papadamou, K., Papasavva, A., Zannettou, S., Blackburn, J., Kourtellis, N., Leontiadis, I., Stringhini, G. and Sirivianos, M. (2020). Disturbed YouTube for Kids: Characterizing and Detecting Inappropriate Videos Targeting Young Children. *Proceedings of the International AAAI Conference on Web and Social Media*, [online] 14, pp.522–533. Available at: https://aaai.org/ojs/index.php/ICWSM/article/view/7320/7174 [Accessed 9 Aug. 2020].

[16] Parab, A., Nikam, A., Mogaveera, P. and Save, A. (2020). *A New Approach to Detect Anomalous Behaviour in ATMs*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9074417 [Accessed 9 Aug. 2020].

[17] Perez, M., Kot, A.C. and Rocha, A. (2019). *Detection of Real-world Fights in Surveillance Videos*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8683676 [Accessed 9 Aug. 2020].

[18] Russo, M.A., Kurnianggoro, L. and Jo, K.-H. (2019). *Classification of sports videos with combination of deep learning models and transfer learning*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8679371 [Accessed 9 Aug. 2020].

[19] Singh, S., Kaushal, R., Buduru, A.B. and Kumaraguru, P. (2019). KidsGUARD. *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*.

[20] Tahir, R., Ahmed, F., Saeed, H., Ali, S., Zaffar, F. and Wilson, C. (2019). Bringing the kid back into YouTube kids. *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*.

[21] Tian, M., Zhu, Z. and Wang, C. (2019). *User-depth Customized Men's Shirt Design Framework Based on BI-LSTM*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/8816528 [Accessed 9 Aug. 2020].

[22] Zhang, Q., Yang, Z., Jiang, Y., Li, H., Han, J., Xu, C., Xu, H. and Xu, X. (2019). *Transmission Lines Scenes Classification Based on Optimized VGG-16*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9066489 [Accessed 9 Aug. 2020].

[23] Zhou, X. and Del Valle, A. (2020). *Range Based Confusion Matrix for Imbalanced Time Series Classification*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9044218 [Accessed 9 Aug. 2020].