Application of Data Analytics To Suggest Gifts To Retail Customers Based On Their Emotions

MSc Research Project
Data Analytics

Sayan Ghosh
Student ID: 21143838

School of Computing
National College of Ireland

Supervisor: Dr. Cristina Hava Muntean

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Sayan Ghosh
Student ID:	21143838
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Dr. Cristina Hava Muntean
Submission Due Date:	15/12/2022
Project Title:	Application of Data Analytics To Suggest Gifts To Retail Customers Based On Their Emotions
Word Count:	8210
Page Count:	20

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:	30th January 2023

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):	

Application of Data Analytics To Suggest Gifts To Retail Customers Based On Their Emotions

Sayan Ghosh
21143838

Abstract

Companies and brands are now constantly expanding their product offerings. This can occasionally make it difficult for customers to choose a product from a large product basket. This problem specifically occurs while purchasing gifts. In this study, customers are given suggestions for different gift categories depending on their emotions using machine learning algorithms. The suggested model uses Random Forest, Convolutional Neural Network, and Residual Neural Network (ResNet50) to recognize different facial emotions. The dataset FER2013 is used to calculate the seven different emotions namely happy, sad, angry, surprise, disgust, fear, and neutral. The dataset contains 35,887 grayscale facial images. Once the emotions have been identified, they will be fed into the classification model together with the dataset of gift category data. The Classification model will assign different categories of gifts to individuals based on their emotions. In this paper, three classification models—Logistic Regression, Decision Tree, and Random Forest—are put forth, and their outcomes are compared. The performance of ResNet50 was better than the other two models for recognizing the facial expressions from the image dataset. The accuracy of ResNet50 was 63.43%. For the gift dataset, Random Forest performed better than the other two classification models, with an accuracy of 79.82%.

1 Introduction

Customer satisfaction is the main focus for all big companies. If the clients are pleased with their goods and services, they are aware that good business will come to them naturally. The businesses strive to provide their customers with a positive purchasing experience. They are doing this by broadening their product lines in addition to improving their services in order to better meet the various demands of their various customers. Even though a wide variety of products is meant to provide customers with more choices, it has been found that many people also find it to be confusing. When presented with numerous alternatives, consumers find it challenging to select a certain product. The idea that buyers should have more options in order to boost sales numbers is rejected by (Kurien et al.; 2014). They have made an effort to examine consumer behavior and the choices that consumers make while purchasing any item. They came to the conclusion that fewer selections make it simpler for a buyer to purchase a product. Consumers are frequently confused by more options, which ultimately prevents sales of the products. In light of this, the researcher suggests creating an application that will suggest various gift categories to individuals based on their feelings and emotions.

Although various studies using facial expression recognition have been produced in the past, it was found that few of them have truly made a contribution to the field of marketing by attempting to address a problem that will aid consumers in making product purchases. This prompted the candidate to carry out additional research and review earlier studies that focused on identifying facial expressions. Most of the earlier studies describe the methods and various models they used to identify emotions in facial images. Many of them have also emphasized the practical uses of their facial recognition models, namely in the realms of medical and security services. The candidate was inspired by this to think of a suggestion that could aid customers in purchasing a product, particularly a gift for themselves or for others. The model will be able to suggest gifts for the customers just by taking a picture of their faces and identifying their emotions. The business sector will greatly benefit from this project, which will not only help to improve customer satisfaction but also enable businesses to reach out to customers with their entire line of goods. The project has taken into account seven main emotion types. They are: happy, sad, disgusted, afraid, furious, surprised, and neutral. To recognize the facial expressions from the image dataset Random Forest, Convolutional Neural Network, and Residual Neural Network (ResNet) have been used for this research. The results of all three models are compared and the one with the best accuracy is considered further. Following the extraction of the emotions, the dataset containing various gift categories and the extracted emotions will be fed through a classification model. Three classification models—Logistic Regression, Decision Tree, and Random Forest—are presented in this paper. This study compares the performance of three classification models and finally selects one of them based on their results and performance. We will receive the required output from the classification model by mapping the gifts (from the gift dataset) to the appropriate emotions. The recommendations for gifts that they might purchase based on this output will then be of benefit to the customers.

1.1 Research Question and Objective

1.1.1 Research Question

How can we use deep learning and machine learning approaches to recognize the emotions of retail customers, in order to give them additional options when selecting gifts from different categories of products?

1.1.2 Research Objective

- Examining and evaluating the research papers in the area of facial expression recognition.
- Images are pre-processed and transformed as part of the model's data preparation.
- Model building and evaluation for the image dataset.
- Prepare Gift dataset.
- Model building, evaluation and comparison for the gift dataset.
- Testing and deployment of the final model.

• Review and comparison of models that have been put into use.

2 Related Work

The author tries to provide a review of earlier research in facial expression recognition in this part. The many methods for identifying facial expressions and their practical applications are the main topics of the following subsections.

2.1 Different techniques used for Facial expression Recognition

(Gill and Singh; 2021) focuses on deep learning techniques to understand the emotions of human beings using facial expressions. Six different emotions namely as happy, sad, angry, surprise, bore, and disgust were taken under consideration. The facial expressions are recognized by applying the convolutional neural network. The model obtained an accuracy of 93% when compared with the existing techniques of facial expression recognition. Sometimes the same kind of information is processed while recognizing the expressions of human faces. It becomes quite difficult to trace back this kind of error. This problem was not tackled in this paper. One of the positives of this paper was that the authors did not rely on a single dataset. Rather they worked with three different datasets to come up with the final model. The Extended Yale Face Database, LFW, and the Google Facial expression comparison dataset were the three datasets used for this research paper. Another paper (Munasinghe; 2018) proposes to identify facial emotions using Random Forest. In the beginning, face landmarks are recognized from all the images. The calculation of a feature vector is done using these face landmarks. Total sixty-eight facial landmarks were identified initially, but since not all of them influence how facial emotions change, just a subset of them is taken into account. A model that was trained using the iBUG 300-W dataset is used to identify the face landmarks. The emotions are then determined by utilizing the Random Forest Classifier. The final model classified the emotions with an accuracy of 90%. These days, neural networks are widely used to recognize images. However, the author chose to employ Random Forest instead of neural networks to recognize facial emotions and was able to get a respectable accuracy rate. Although the findings are satisfactory, this particular study only included four different emotion types namely anger, happiness, sadness, and surprise.The elements of both speech and facial expressions are used in this paper (Cai et al.; 2020) to identify emotions rather than relying solely on one mode. Utilizing long short-term memory (LSTM) networks and convolutional neural networks, the speech-related expressions were retrieved. The model was tested on the Interactive emotional dyadic motion capture (IEMOCAP) database. Using deep neural networks, the characteristics of both speech and facial expressions were merged. They may have used other modalities (such as text) in their model, rather than just two, to support their issue of multi-modal emotions. The total recognition accuracy of the suggested model was improved by 10.05% and 11.27%, respectively, over the single modal of speech and facial expression. Applying Deep Learning principles, this study suggests a multimodal sentimental analysis. In addition to the speaker's face expression, they have taken into account the speaker's verbal and vocal qualities for this aim. To achieve their goal, (Shrivastava et al.; 2018) combined facial and voice expressions to acquire the intended outcome. They have succeeded in developing an information processing model that is based on the human nervous system with the aid of deep learning techniques. They were able to successfully complete the sentimental analysis by extracting the facial emotions, together with the verbal and audible components, using all three methodologies. They were able to get findings from the vocal and verbal analysis module with an overall accuracy of up to 91.80% (training) and 88% (testing), whereas the results from the facial expression analysis module had an accuracy of up to 93.71% (training) and 92% (testing). In this study (Yen and Li; 2022), a series of experiments were performed on the FER2013 dataset. A total of five different models were used to conduct the study. These five models are ResNet-50, Xception, EfficientNet-B0, Inception, and DenseNet121. Other than FER2013, they have also used AffectNet dataset to determine the best models. This paper has also used different data pre-processing techniques and training methods. With regard to AffectNet, the accuracy of the ResNet-50, Xception, EfficientNet-B0, Inception, and DenseNet-121 models improved by 8.37%, 10.45%, 10.45%, 8.55%, and 5.47%, respectively. These models' respective accuracy improvements on FER2013 were 5.72%, 2%, 10.45%, 5%, and 9%. The paper should have also emphasized some of the models' real-world uses. (Zhou et al.; 2019) proposed Resnet-18 network for recognizing facial expressions using the FER2013 dataset. They observed that there was still an opportunity for improvement in the accuracy of the FER2013 dataset. Using the Resnet-18 model, it was found that a significant portion of the model's parameters was vulnerable to over-fitting, which reduces the accuracy of face recognition. Here the average pooling layer of Resnet-18 was replaced by the global average pooling layer along with two convolution layers. Though the accuracy was improved by 1.49%, but the accuracy for the emotion types anger, fear, and sad was not encouraging.

2.2 Application of Facial Expression Recognition in the Real World

One of the most common methods of getting customer feedback is to have individuals fill out questionnaires, which frequently run into the issue of customer's willingness to participate. Instead of filling out those lengthy questionnaires, this paper (Chen et al.; 2020) attempted to solve the problem by identifying the customer's emotions from their expressions in the video. For this research work, the data was collected from real museums and formed a dataset named MuseumDS. The researchers utilized a model based on a bidirectional long short-term memory network (BiLSTM) to capture the various emotional changes. The study's findings enabled precise identification of the customer's cumulative emotions. The comprehensive facial emotional expression throughout time is analyzed using the VBA model in this study in order to gather feature data for classification by images with a predetermined number of frames. The optimal number of frames for the model is ten. Though the quality of the emotion recognition needs to get improved as there are different noises such as the lighting environment, angles, and others in the pictures. The major goal of (Liu et al.; 2019) was to recognize masked individuals. People sometimes cover their main facial features with masks or heavy makeup to conceal their identities. The primary alterations in the facial muscles and the various facial motions, which primarily happen when speaking, are substantially captured in this study. Spiking Neural Networks have been utilized in place of Convolutional Neural Networks, and they employ the dynamic movements of faces as their primary input. They are originally coupled with deep learning architectures before being applied to the Spiking Neural Networks. The experiments were conducted on a video-based disguise face database (MakeFace DB), achieving more than 95% classification rates under various realistic testing conditions. However, they haven't really addressed the cases where they have complicated changing backgrounds, which is one weakness in this project. (Khare and Bajaj; 2020) has applied convolution neural networks in their research and for the purpose of identifying emotions, electroencephalogram (EEG) signals are used. The EEG recordings from 20 students, with a mean age of 23, were used as a dataset. The suggested method uses a time-frequency representation to turn the filtered EEG data into an image. To generate visuals from time-domain EEG signals, a smoothed pseudo-Wigner-Ville distribution (SPWVD) is used. They have combined convolution neural networks with a two-dimensional smoothed pseudo-Wigner-Ville distribution function of time and frequency. According to their analysis, AlexNet, one of the varieties of convolutional neural networks, outperforms ResNet50, VGG16, and VGG19 in terms of performance and training time. An accuracy score of 90.98% was obtained using AlexNet. One of the shortcomings of this model is that the size of the dataset is very small. It would be interesting to see how the model performs if more EEG recordings are taken into consideration. (Naas and Sigg; 2020) discussed that perceiving the emotions of the customers can have a positive impact on overall sales performance. They suggested an audio-visual emotion identification system that can distinguish between the following eight emotions: neutral, calm, sad, happy, furious, afraid, startled, and disgusted. This system is intended to support emotional perception in buyer-seller exchanges. Working with two separate datasets, Surrey Audio-Visual Expressed Emotion (SAVEE) and Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), they constructed a model that they used to determine the correctness for both audio and video. They were effective in achieving accuracy rates for video emotion recognition of 97.13% and 97.77% and audio emotion recognition of 62.51% and 68% on the same datasets respectively. Finding out consumer satisfaction was the main motivation for performing this survey. This model was tested in real-time conversations. The convolutional neural network was employed for the classification of the data. In today's driver warning systems, facial emotion recognition plays a significant role. Predicting suspicious activity involving law enforcement and security agencies can also be helpful. The goal of this study (MADUPU et al.; 2020) is to use the concept of facial expression detection to predict a person's propensity toward suicide. The six basic emotions have been the focus of the majority of facial recognition research; however, this paper has focused on the mixed emotions that individuals experience. These mixed emotions can help to accurately define a person's state of mind and characteristics. Convolutional neural networks have been applied to the features taken from the Speeded Up Robust Features (SURF). Although they have a 91% accuracy rate, this work can be expanded to a larger dataset because the results in this research were only attained with a small dataset of 200 samples.

Analysis of past studies and research papers revealed that the concept of emotion detection hasn't been applied frequently to create marketing solutions that can help customers. The researcher has therefore put up a notion that would enhance consumers' shopping experiences by allowing them to choose between various items based on the emotion patterns that were recorded using the facial expression detection technique.

3 Methodology

The methodology followed in this research is the Cross Industry Standard Process for Data Mining (CRISP-DM). The CRISP-DM has six sequential phases which are described below. A pictorial form of the methodology has been shown in Figure 5[1]

1. Business Understanding

The researcher explains how the project will contribute to the needs of the business throughout this stage. Due to the variety of products, it has been observed that customers occasionally struggle to choose a product when buying any kind of gift. This issue was addressed by developing a method to suggest various gift categories to buyers based on their emotions.

2. Data Understanding

This project deals with two different datasets. The first dataset is the FER2013, which has 35,887 grayscale images. The dataset was obtained through Kaggle[2], a website that provides access to a variety of datasets. In order to recognize facial expressions, the FER2013 dataset is employed. The second dataset is titled "Gift dataset," and it contains all the information pertaining to various gift categories. The dataset was produced using Mockaroo[3], an online tool for producing data. As Mockaroo only allows us to create datasets with 1000 rows, the researcher created five separate datasets that the researcher later aggregated to create one larger dataset. There are 5000 rows in the final consolidated dataset.

3. Data Preparation

In this phase, the data is pre-processed and organized before feeding it into the final model. In the first dataset (FER2013), used for facial expression recognition the researcher had to deal with grayscale images. Techniques including data augmentation, batch normalization, flattening, padding, and MaxPooling are used to pre-process the images. The images are rotated, resized, and shuffled during the image augmentation. The data preparation approaches are different for the three models since the researcher utilized three distinct models to identify the facial emotions from the image collection. The implementation section has a detailed discussion of every pre-processing step used in the three models. Given that the Gift dataset is made up of synthetic data that the researcher created himself, pre-processing is not necessary.

4. Modelling

Random Forest, Convolutional Neural Network, and ResNet50 (Residual Network) have all been developed for facial emotion recognition. The researcher has further implemented three models for the Gift dataset: Decision Tree, Logistic Regression, and Random Forest. The Implementation Phase has covered every aspect of the Modelling phase in detail.

5. Evaluation

To evaluate the model performances, Accuracy and Loss percentages have been used as the two main evaluation matrices.

6. Deployment

The deployment stage of the study is where the researcher demonstrates how the full model actually functions in real-time. The model is able to suggest several categories to

[1]Figure 1 has been created by the researcher using- URL: https://app.diagrams.net/

[2]URL: `https://www.kaggle.com/datasets/msambare/fer2013`

[3]URL: `https://www.mockaroo.com/`

clients based on their emotions.

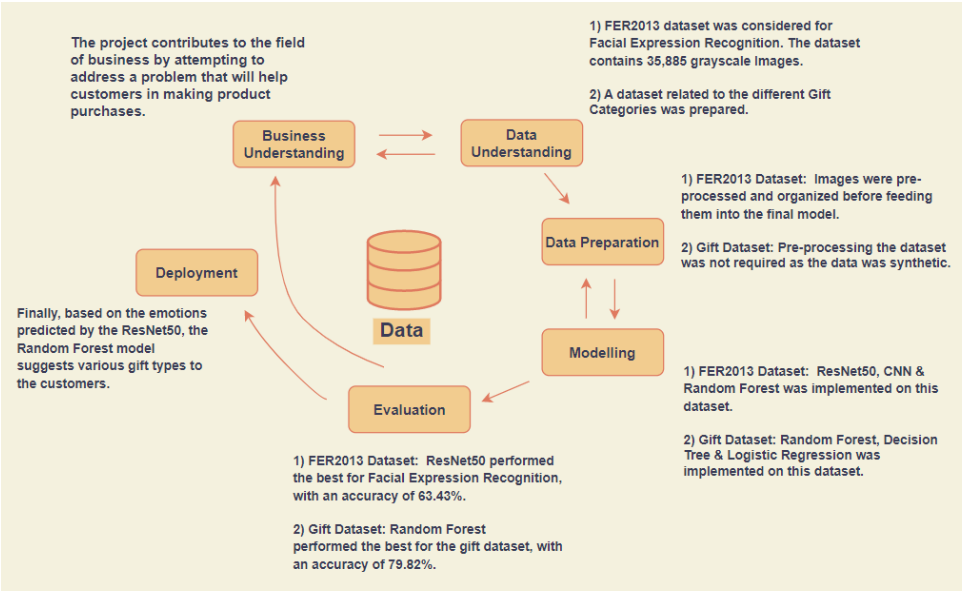

Figure 1: Project Methodology

4 Design Specification

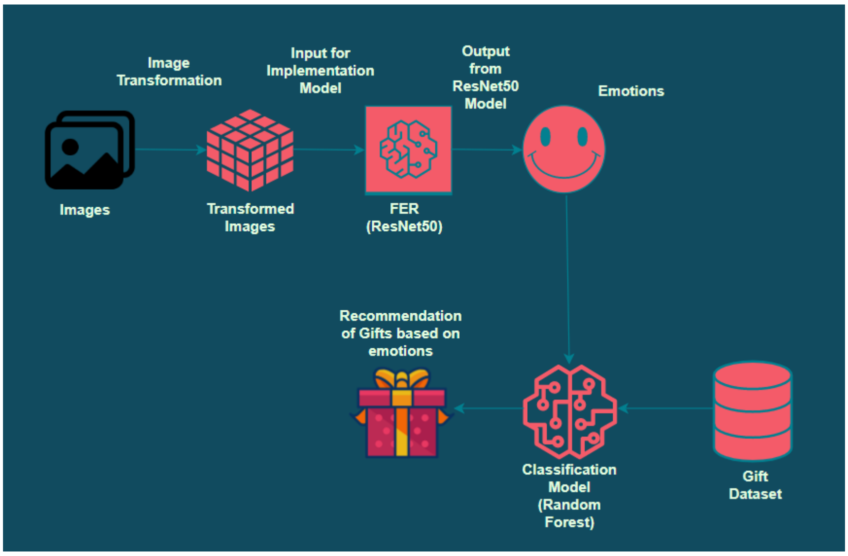

Figure 2: Project Flow Design

Figure 2[4], shows the process flow diagram of the project. It reflects the steps which have been taken for this particular research project. The diagram shows the whole process of recommending different gift categories to the customer, based on the emotions captured

[4]Figure 2 has been created by the researcher using- URL: https://app.diagrams.net/

by the model. The images taken from the image dataset are initially processed and transformed in order to feed them to the ResNet50 model. The ResNet50 model will predict the emotions from these images and send them to the classification model. The classification model used in this project is Random Forest. Along with the emotions, a gift dataset is also passed to the Random Forest model. Once the images and the gift dataset are passed through, the classification model will recommend different categories of gifts to the customers. All the steps are described in detail in the later sections of this report.

5 Implementation

The FER2013 dataset consists of 35,887 images of 7 different emotions: anger, neutral, disgust, fear, happiness, sadness, and surprise. Once the dataset is downloaded, the researcher can see the images are already segregated into two different folders: Training and Validation. 80 percent of the data are in the Training folder and the rest 20 percent are in the Validation folder. There is total of 28,821 images in the Training folder and 7,066 images in the validation folder. Inside each folder, the images are further divided into seven different folders based on the seven different emotions. A computer with a 64 bit x 86 multi-core CPU, 8 GB of random-access memory (RAM), and more than 250 GB of hard disk space is the minimum hardware requirement for carrying out this research. Python and its associated libraries were used to perform the coding.

5.1 Facial Expression Recognition

5.1.1 Convolutional Neural Network (CNN)

Pre-processing and Augmentation:
Initially, the size of the images was specified using the figsize attribute of Matplotlib library in Python. The height was set out as 12 and the width as 20. For the pre-processing of the images, the researcher uses ImageDataGenerator to transform the images in a way that will further result in fast processing. ImageDataGenerator is a class of the Keras open-source Python library. To receive the input of the original data, transform it randomly, and then create an output resultant that only contains the newly altered data, ImageDataGenerator is used. A batch size of 128 images was taken. This means that in each training iteration, the ImageDataGenerator will then generate 128 images. The reason for doing this is as the total number of images is quite large, so the researcher process batches by batches and sends them serially one by one.

In the next step, the researcher created the train_datagen and validation_datagen. Image augmentation is done so that if any new image is considered in the future the model might not be generalized because of over-fitting or under-fitting. To avoid that, image augmentation is performed where the images are rotated randomly by 20 degrees. It is done so that even if any of the images are received rotated or in any other form it will be able to detect it. This helps to capture the actual features from the images. The researcher also re-scale the images. While re-scaling, the researcher normalizes them on the basis of pixels. Zero (0) is the minimum pixel which is denoted white colour and 255 is the maximum pixel which is black colour. Rescaling to 1/255 will convert the pixels in the range [0,255] to the range [0,1]. This process is also called Normalizing the input.

Scaling every image to the same range [0,1] will make images contribute more evenly to the total loss. Other arguments of the ImageDataGenerator also include Image rotation, height_shift_range, and width_shift_range. The height_shift_range and width_shift_range shift the image along the height dimension and width dimension respectively. In both cases, images are moved by 0.1 pixels. Horizontal flip is performed for flipping the images. All these are performed in the train_datagen and in the case of validation_datagen, the images are re-scaled. The researcher builds the datagen object by passing these arguments to the ImageDataGenerator using the Python keyword arguments.

The train_generator is basically creating the train data of images. Here the target size is 56 by 56, where 56 is the number of pixels in the row. We are basically re-scaling the 128-pixel images to 56 pixels horizontally and 56 pixels vertically. This is the resolution selected for the images, so the number of pixels will be less, and the image size will also get reduced. It helps to pass all the images freely through the model. The class_mode is selected as categorical as our data is categorical data. The researcher keeps the shuffle as 'True' so that our model is not biased. To avoid a biased model, images are shuffled in order to get different images of different emotions rather than the model learning only one emotion at once. Doing the same process for validation_generator, except shuffling the images as it is not training the data, instead, it just validates our training results.

The flow_from_directory function of this object should then be used as the next step. When the images are arranged into folders in our Operating system, this technique is applied. This is applied to both the Training and Validation folders.

Training Phase:
As we are dealing with grayscale images (either black or white), the researcher uses Convolutional 2D. The researcher cannot use Convolution 1D and Convolutional 3D. Convolutional 1D is used for the text data and Convolutional 3D is used for RGB images. Here the researcher is adding three layers of Convolutional so that the kernels will be more generalized.

Batch Normalization: Batch Normalization is basically done using neural networks to normalize the weights and accelerates the learning process (Khaireddin and Chen; 2021). ReLU (Rectified Linear Unit) has been used as the activation function. To get a weight assigned to a particular neuron, an activation function is used. Using this, the researcher is basically predicting each artificial neuron. MaxPooling is converting all different dimensional data into single 2D data. Another technique known as Dropout is used for generalization. Instead of making the model biased, the researcher is dropping out 0.25 percent of the data.

Padding: All the images have different sizes. Some will be large, and some will be small. In the case where some images are smaller than the actual image size, extra layers of zeroes to make the model fit for all the images. The way we impute the null values in the case of text data, padding does the same thing in the case of images.

Flattening: The researcher is adding three layers of Convolutional so that the kernels will be more generalized. Once the kernels are generalized, we are flattening them. Till this point, the data was two-dimensional (horizontal and vertical layers). Here the researcher is converting the data to fit it in a normal data frame, as our normal data frame is one-dimensional. Flattening creates a matrix (or a NumPy array) and then passes it to the dense layer. Whatever weights we have before flattening, have the actual features of the images. The weights keep on passing from one layer to another. To be precise, the output of the final convolutional layer is the actual input of our neural network. All the feature data are stored in the final output of the convolutional layer (in this case the fourth layer). By the end of the fourth layer, we have fine feature weights of the images or the categories.

Dense Layer: The dense layer is basically the perceptron. Perceptron is the basic neural network. The data is passed to the perceptron. Here the researcher is adding two layers of dense. 256 in the first dense layer and 512 in the second dense layer are nothing but the neuron outputs. So, the researcher passes 256 to 512 output or classes. After that, it is passed to the dense layer with nb_classes. The nb_classes are the number of output classes. There are seven different classes (emotions) in our dataset. Therefore, in this case, nb_classes will be 7. In the end, it is passing 512 classes to 7 classes.

Activation Function: Softmax has been used as the activation function which is either 0 or 1. The Softmax function is frequently employed as a neural network's final activation function to normalize the output to a probability distribution over expected output class. It assigns 0 to 6 numbers to all seven classes.

Compilation:

Optimizer: Adam has been used as the optimizer. Adam is the abbreviation for the Adaptive Movement Estimation method, which is a variation of gradient descent. Adam originally has a learning rate set, but it will adjust to the real learning rate and seek out the most effective one to find the local minima or global minima (Khasanov et al.; 2021). During the training process, the learning rate was set to 0.0001. Compared to other optimization techniques, the Adam optimizer typically produces better results. It computes more quickly and requires fewer tweaking settings. As a result, Adam is suggested as the standard optimizer for image processing.

The categorical_crossentropy is the loss function and Accuracy is the Metrics. The Activation function is used for prediction and the loss function is to identify whether the process is moving in the right way or not. Each iteration or each backward and forward propagation should reduce the loss.

During the training process, the learning rate was set to 0.0001 and the batch size was 128, in order to enhance the robustness of the model.

Once all these steps are done, then the training data is iterated in order to train our machine-learning model. One iteration is one EPOCH or we can also say that a single EPOCH is the single forward and backward pass through the neural network of the complete dataset. The best accuracy for the model is achieved at 29 EPOCHs.

Model Tuning:

Initially, the researcher tried to test the model using 25 EPOCHS. To further improve the accuracy, the researcher implemented a total of 35 EPOCHS and monitored the total loss and accuracy. As per the results, the model has attained the best validation accuracy at 29 EPOCHs. The Validation accuracy and loss percentages of the different EPOCHs are mentioned below in the Table 1.

Table 1: Model Tuning

No. of EPOCHs	Validation Accuracy	Loss
25	55.54%	1.2497
29	59.46%	1.2098
35	56.78%	1.1596

5.1.2 Random Forest

To recognize the facial expressions from the image dataset the researcher also tried using Random Forest, which is a machine-learning algorithm. In most cases, deep learning methods are typically used for image processing because they employ neural networks, to directly learn usable interpretations of features from input. But in this case, the researcher also tried a machine learning model along with the deep learning techniques in order to monitor the performance of both the techniques.

Pre-processing:

Initially, all the images are re-sized. Images are cells of data with values ranging from 0 to 255. The images (0 to 255 pixels) are converted into a CSV File. The number of columns depends on the resolution of the image. For e.g., if the image resolution is 48 x 48 then, the 0 to 255 pixels will get converted to 48 different columns. Then the class names of each image are captured. The re-sizing is done for the training and testing data.

```
Out[6]: array([0.59231373, 0.59090196, 0.58666667, ..., 0.70980392, 0.71733333,
               0.71984314])
```

Figure 3: Array Output for Random Forest

Figure 3, shows an output of array. This array is normalized pixel weights for each image. If it is a two-dimensional (2D) matrix, it would be difficult to convert it into a CSV, therefore we are flattening it and bringing everything into one single dimension. By referring to the above example of image resolution of 48 x 48 we can explain that a two-dimensional (2D) matrix will have 48 images vertically and 48 images horizontally. So, there will be a matrix of 48 x 48. But it is not possible to fit the 48 x 48 matrix in a dataset. Therefore, this 48 x 48 matrix is converted into a 48 x 1 matrix. Then each image will fit in one single row.

Just for self-check, the total number of images present in each class is counted, as shown in Figure 4. After that, we make sure that everything is converted into a NumPy array. Then the array is passed into the train_test_split function. The train_test_split function takes the array and separates the test data and train data at a 70-30 split ratio. Test data contain 30% and the rest 70% is test data. The researcher uses random_state to re-check whether we get the same data for training that we had used earlier. The number 69 given to the random_state is nothing but a normal seed value. Shuffling of the images is also performed to make sure that our model is unbiased.

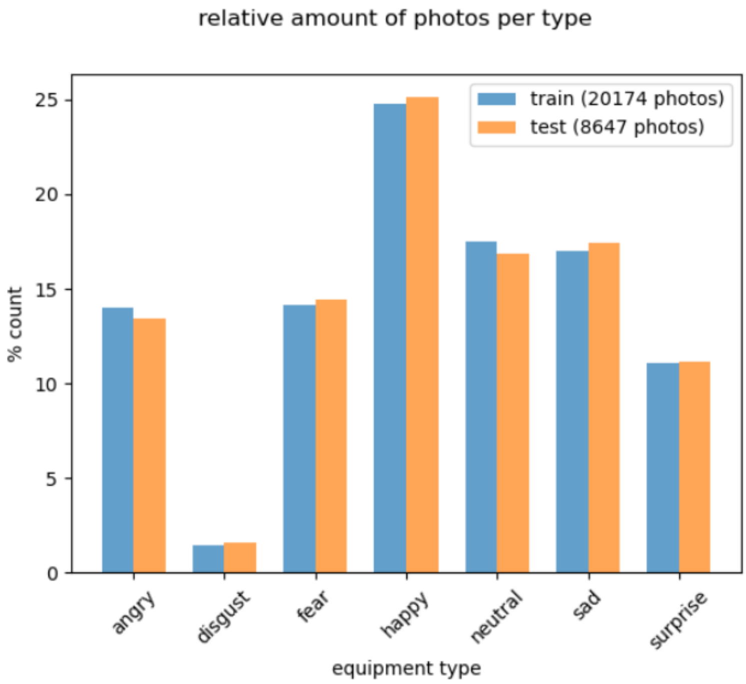

Figure 4: Images in each Class

Model Building:

RGB2GrayTransformer() is used in order to convert RGB images to grayscale. If any image is coloured then it will get converted to grayscale using this function. The ensemble model is used to overcome overfitting or underfitting. If there are repetitions in the data then overfitting occurs. To overcome overfitting, Bagging techniques are used and to take care of underfitting Boosting techniques are used. In our case, the researcher generalized our model so that it doesn't overfit. In this case, the researcher has built 300 decision trees (n_estimators = 300), where each tree is built only on a sample of data. Basically, we take a ranking out of these 300 trees. Based on the highest ranking, a particular class is selected. This is the way the whole model is generalized.

5.1.3 Residual Network (ResNet50)

Residual Network (ResNet) is an effective Convolutional Neural Network (CNN), that has more than 150 layers and aids in resolving the vanishing gradient issue. The vanishing gradient tends to happen as CNN gets deeper, which is bad for the network's performance. When a gradient is backpropagated to prior layers, vanishing gradient problems arise and a very small gradient is produced. The "skip connection" feature of ResNet makes it possible to train numerous deep layers without worrying about vanishing gradient problems. With skip connections, as the name implies, parts of the neural network's layers are skipped and the output of one layer is used as the input for the layers that follow. In ResNet the weights of the previous layer get added to the next (alternate) layer so that the output is generalized and layer by layer the weights get refined.

The researcher is not constructing the ResNet independently. We are importing the pre-trained ResNet50 model and working on it. After importing the ResNet50 the researcher is using the weights as it has already captured a lot of features in it. Simply rearranging the weights for our dataset is all that is being attempted here.

The initial pre-processing and augmentation part is similar to the one the researcher already performed in our first experiment (Convolutional Neural Network), except for the colour_mode in train_generator and validation_generator. The colour_mode in the train_generator and in the validation_generator is mentioned as 'RGB' instead of 'grayscale', in order to match the array size.

Model Building:

Since an input layer already exists, the researcher set include_top to False and are eliminating the top layer. ImageNet has been used as 'weights' because the ImageNet is very detailed as it contains more than 14 million images. A sizable visual database called ImageNet was created mainly for the purpose of use in research on visual object recognition software. Basically, the ResNet model is trained on this ImageNet data. Therefore, the ResNet50 contains all the detailed weights on it. Therefore, what we are doing is capturing all of these weights and asking the ResNet50 model (that has already been trained) to provide the weights for the model that we are training. We just need to rearrange the weights for the dataset we have. The model may not capture enough features if we attempt to train it from scratch. The information might or might not be sufficient. So instead of assigning random weights, we are initializing with the actual feature weights from different images. Therefore, we are initializing the weights with the known weights that have proven capacity or capability to predict anything. However, since it is not generalized to our dataset, we are initializing weights that are as close as possible to our dataset and once the backpropagation is done this re-arrangement will be done faster compared to normal neural networks.

Table 2: ResNet50 Parameters

S.No.	**Parameters**	**Value**
1	Optimizer	ADAM (Adaptive Movement Estimation)
2	Loss	Binary Cross Entropy
3	Metrics	Accuracy
4	Activation Function	Softmax

Table 2 shows all the other parameters which were considered for building the model process.

5.2 Gift Dataset

The researcher constructed the gift dataset on his own using the internet website Mockaroo. As Mockaroo permits us to build a dataset of just 1000 rows, so we produced five independent datasets and combined them later to form a bigger dataset. The columns provided for the gift datasets are Gift Categories, Emotions, Gender, Occasion, and Price. Cakes, Flowers, Personified Gifts, Fashion & Lifestyle Gifts, Jewellery, Home & Living Gifts, Toys & Games, Gourmet, and Plants are the different gift categories that were considered for the dataset. Occasions were mainly divided into three different types: Special Events, Festivals, and Casual. To test whether there is any variation in the data, gender was included. From the GDPR point of view, even age was not considered for the gift dataset. In any case, the researcher intends to eliminate gender for later uses. The gender was retained for the research project solely to add more features and observe how the model varies or changes. It was believed that gender could lead to many layers of feature splitting or differentiation since women might favor certain categories of gifts while men might prefer other types of gift categories. After the model was constructed, it was discovered that gender was not assigned much feature importance. In the future, other features will be taken into account and gender will be removed from the list of features. In place of gender, regions might be thought of as one of the traits. No specific region name will be used; instead, masked regions (in form of categories) can be taken into consideration for this particular work. Honestly, no steps were taken to avoid gender bias prediction. However, the researcher considered employing the Synthetic minority over-sampling technique (SMOTE) to avoid the imbalance but was unable to do so due to time constraints. SMOTE takes the under-sampled data and creates synthetic samples out of it. Basically, it re-creates the minority class data. Due to time constraints, the researcher considered forgoing the SMOTE method and simply taking the data as it is.

Once the gift dataset is prepared, we are then training it. We do not have to work on the pre-processing part for the gift dataset reason being it is synthetic data, that has been created by the researcher himself. Before training the model, one-hot encoding and label encoding are performed. In one-hot encoding, the strings get converted to numerical form (mainly 0 and 1) and in label encoding, the strings are converted into numbers. The One hot encoding which is performed on the training data is also performed in the test data otherwise the whole classes will get interchanged, and it won't work. Then Pickle files are created for each of the one-hot encoders. Pickle involves turning a Python object into a byte stream so that it can be stored in a file or database. For gender and occasions, we are initially doing the one hot encoding and then pickling and dumping those into OHE_Gend and OHE_OCC files respectively. Once the pickle files are created, these are then sent to the pre_process folder. The pre_process folder is created in the same directory where all the test, train, and validation folders are present. Label encoding is performed for the emotions (le_Emo) and for the output (le_op). Label encoding is also carried out for emotions as there are seven different types of emotions. So, after label encoding, all seven emotions will be numbered from 0 to 6. We are doing label encoding for the output as well so that if we give the class number, in order to get the name of the class accordingly. In short, if we provide the numbers, the model will give us back the name of the label (emotion). This is the reason we are also saving the output in the pre_process folder. The same steps are followed for the label encoders as well i.e., pickling the label encoders and dumping them into the pre_process folder.

Models Used:

In this phase, the researcher has tried out three different models. These models are Logistic Regression, Random Forest, and Decision Tree. Based on the accuracy percentages of all three models the researcher decided to go with the Random Forest, which gives a validation accuracy of 79.82%. The researcher is creating another folder in the directory and named it 'models'. In the 'models' folder, we are pickling and saving all four models: Random Forest, Decision Tree, Logistic Regression, and the ResNet50. The ResNet50 model is saved because it performed the best while recognizing the facial expressions from the image dataset.

5.3 Testing & Deployment

For predicting the researcher created one function called DL_model. In the codes, using the keras.models.load_model, the researcher directly load the trained ResNet50 model. So, the researcher need not re-train it again. The DL_Pred is where the actual model is predicted. To do the predictions, the researcher already did the pre-processing of the validation data. Likewise, the researcher re-created the same for Testing. Similarly, the researcher created test_datagen and test_generator using the ImageDataGenerator. Then the researcher is creating a dictionary inside which we are mapping with all the seven emotions namely anger, disgust, fear, happy, neutral, sad, and surprise along with the numbers from 0 to 6. The output would be something random where it will be in the form of the probability distribution of each class (emotions). For e.g. After passing an image through our model, if the output shows angry- 10%, Sad- 20%, fear- 50%, disgust- 10%, happy-1%, neutral-4%, and surprise-5%. Therefore, considering the highest probable factor, which is fear (50%). Basically, the emotions which have the highest probability percentage is chosen. To get the maximum probable outcome the researcher is using 'argmax'. Argmax is a NumPy function that identifies the argument in a target function that yields the highest value. The most typical application of Argmax in machine learning is to identify the class with the highest predicted probability. One more function has been created to read the pickle files. To read a pickle file we just need to provide the file name of the pickle.

Data Preparation for User Input:

For the data preparation, we are taking inputs from the user. Price, Gender, and Occasion are the three inputs that the user will be providing. Emotions are directly taken from the predictions made on the images present in the test folder. The price (in Euros) is converted into integer format. Gender and Occasion are read in string format and then converted to numerical using the one-hot encoding.

The data is now ready, and the model needs to recommend the gift categories based on these user inputs. To do this, first, the pickle file of the random forest model (as it gave the best accuracy results while training the gift dataset) which is present in the 'models' folder is loaded. Secondly, to get the class names the output label encoding (le_op) pickle file is loaded, which is present in the pre_process folder. Lastly, this prepared data is saved into the data frame (df) and passing it directly to predict function (Gift_cal_preds). Simultaneously the researcher also has to create a Test folder in our computer and dump random images into it and pass those images into our model. In turn, the model will return different options of gift categories based on the type of emotions predicted by the model. The output of the final gift prediction is shown below in Figure 5.

User Input, Testing Outcome:

Initially, three random images were passed into the Test folder, located in the directory. The emotions of all three images were recognized by the ResNet50 model. For each of the three individuals whose images were passed through the model, information regarding their gender, their budget (Price) for the gift they want to buy, and the occasion

```
Enter Price:123,242,158
Enter Gender (Male\Female):Female,Male,Male
Enter Occasion:Special Events,Casual,Festivals
Found 3 images belonging to 1 classes.
1/1 [==============================] - 1s 1s/step
   Price (Euros) Emotions  Female  Male  Casual  Festivals  Special Events
0            123        0     1.0   0.0     0.0        0.0             1.0
1            242        2     0.0   1.0     1.0        0.0             0.0
2            158        3     0.0   1.0     0.0        1.0             0.0

array(['Fashion & Lifestyle Gifts', 'Gourmet', 'Cakes'], dtype=object)
```

Figure 5: Output of the final Prediction

for which they want to purchase the gift were captured (inputted). Based on these input data it can be seen in the Figure 5, the model has recommended Fashion & Lifestyle Gifts as the gift category for the female, Gourmet for the first male, and Cakes for the second male.

6 Evaluation & Discussion

All of the significant findings and analyses from the implementation phase are presented in this section. The entire research has two parts. The first part is related to the evaluation of the models used for facial expression recognition and the second part consists of the models used for the 'Gift' dataset.

6.1 Facial Expression Recognition Models

This section has covered the performance of CNN, Random Forest, and ResNet50 for image processing. The accuracy score has been used to evaluate each of the models. The evaluation is done to determine how well the model can distinguish between various emotions from photos of facial expressions.

6.1.1 Convolutional Neural Network (CNN)

The model was run for all total 35 EPOCHs. Accuracy has been used here as the evaluation metric. The accuracy for validation data is 59.46%. The get_best_epoch function helps in getting the best validation accuracy score for a particular epoch. As per this function, the best Validation Accuracy is for epoch 29. The validation loss at 29 EPOCH is 1.2098. The training loss and validation accuracy of the model is illustrated below in Figure 6. In the figure, it can be seen that the validation accuracy tends to decrease after 29 EPOCHs and the loss is minimum for the 35th EPOCH.

6.1.2 Random Forest

Apart from using deep learning techniques, the researcher also thought of implementing a machine learning algorithm for recognizing facial expressions. The Random Forest

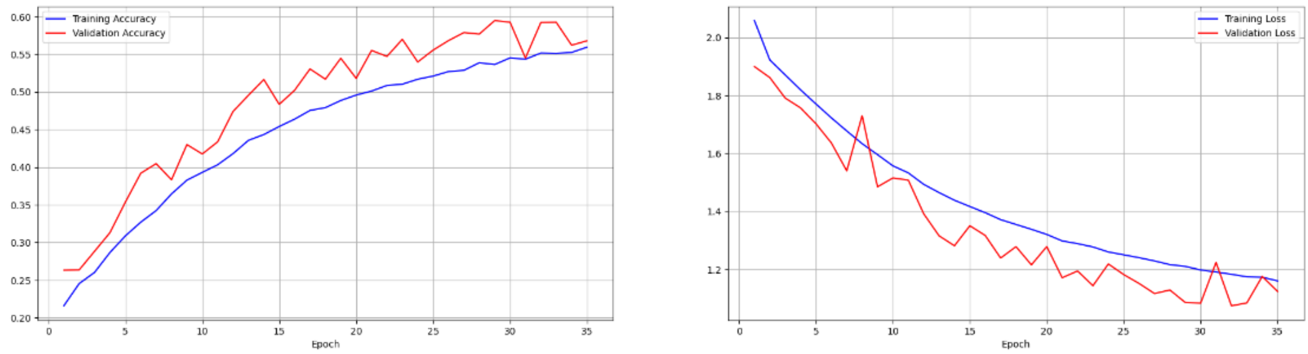

Figure 6: Loss and Accuracy Graph for CNN

algorithm has been used for the same purpose. The accuracy score for this model is the least among all three models. The accuracy percentage for this model was 47.12%.

6.1.3 ResNet50 Architechture

Same as CNN, total 35 EPOCHSs were run for the Resnet50 model. As per the get_best_epoch function, the best Validation Accuracy is for epoch 24. The model has achieved a validation accuracy of 63.43%. The validation loss at 24 EPOCH is 0.335. The training loss and validation accuracy of the model is illustrated below in Figure 7

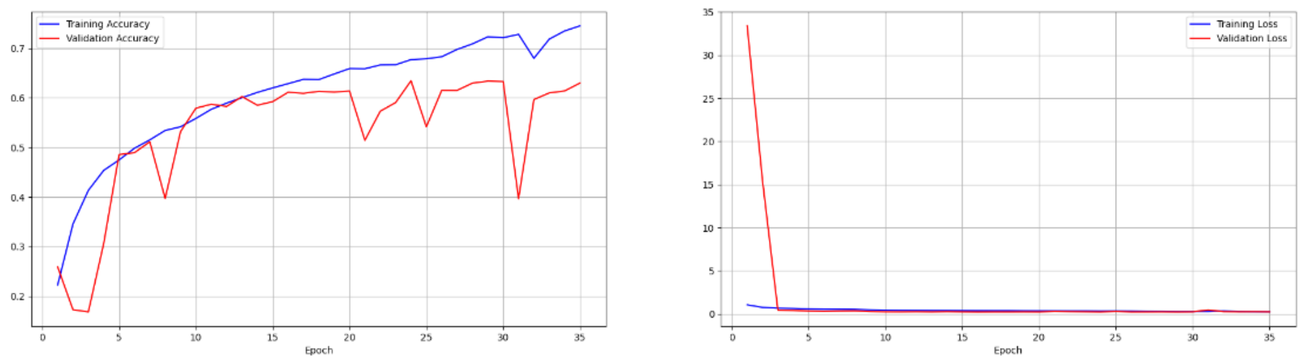

Figure 7: Loss and Accuracy Graph for ResNet50

6.2 Gift Dataset Models

Table 3: Model Comparison Table

Models	Training Accuracy	Val. Accuracy
Random Forest	79.08%	79.82%
Decision Tree	79.08%	77.89%
Logistic Regression	12.72%	12.50%

Three different machine-learning models were implemented on the Gift dataset. The training and validation accuracy of all three models are shown in Table 3. It is clearly seen that the performance of the Random Forest and the Decision Tree are good for the gift dataset. Due to a slightly better validation accuracy, the Random Forest model has been chosen as the best model and has been used for the final stage of deployment, where the customers are recommended various categories of gifts based on their emotions, despite the fact that there isn't much of a difference in accuracy between the Decision Tree and Random Forest models.

For the Random Forest Model, the researcher tuned all the following parameters mentioned below, using the GridSearchCV technique. GridSearchCV is a method for looking up the best parameter values in a grid of parameters.
1) bootstrap 2) max_depth 3) max_features 4) min_samples_leaf 5) min_samples_split 6) n_estimators
However, none of the aforementioned hyper-parameter tunings produced superior outcomes, therefore making the decision not to use all of them. In the end, it was agreed that the n_estimators should be the only ones used, and the other parameters should be dropped because it was also quite time-consuming to tune additional hyper-parameters, which in any case were not showing any evidence of delivering superior results.
For the Decision Tree Model, the researcher tuned all the following parameters mentioned below, using the GridSearchCV technique.
1) max_depth 2) min_samples_leaf 3) min_samples_split 4) criterion
Similar to Random Forest, it was chosen not to include Decision Tree's hyper-parameter tuning in the final model because it did not produce any positive results. Parameter tuning was not tried for Logistic Regression because it was not giving good results. So, it was decided to tune the parameters only for the Random Forest and the Decision Tree models.

ResNet50 and Random Forest are the two models that perform the best for both the facial emotion recognition dataset and the gift dataset, respectively. Both models, as stated and demonstrated in the "Testing & Deployment" part of the Implementation, have successfully accomplished the primary goal of the research, which was to forecast various gift categories for the consumers. Additionally, since many earlier studies (Gill and Singh; 2021) only addressed six different emotion types, dealing with seven emotion classes, particularly on a large image (containing 35,887 grayscale images) dataset, was a bit more challenging. Nevertheless, the researcher feels that a few more Evaluation metrics could have been used to further verify the performance of the models.

7 Conclusion and Future Work

The majority of businesses and brands are continually increasing the range of products they provide. This is done, to give their clients more purchasing possibilities. But clients often face difficulties when trying to choose a particular product from the vast array of available choices. As a result, the customers are not just confused, but many products are left unsold, which also hampers the overall sales of the company. In order to cater to this problem, the research has developed a method to suggest several gift categories to customers based on their emotions. To achieve this, firstly a dataset of images is used to identify the emotions of various people. Facial expression recognition has been implemented using the ResNet50 model. The model's accuracy was 63.43%. Secondly, the Random Forest successfully predicted the gift dataset, with an accuracy of 79.82%. Finally, based on the emotions predicted by the ResNet50, the Random Forest model suggests various gift types to the customer. In this report, the researcher has successfully demonstrated the process of recommending different gift options based on different user (customer) inputs and emotions. Consequently, the primary goal is achieved.

A mobile application that allows users to receive product recommendations based on their emotions might be created as a follow-up project based on the same idea. Alternatively, the model might also help gift merchants in the future. With the use of this framework, businesses can utilize it to forecast the feelings of customers who visit their stores and give them an additional gift-choice alternative. It is also suggested to improve the current model by utilizing modern facial application technologies. The gift dataset used in the project is synthetic data. It is synthetically generated data. In the future, we plan to do market research and extract proper data. The models are not well-tuned. For the actual dataset, the models need to be re-tuned. We need to collect more samples and add more dimensions to the data.

8 Acknowledgement

I want to express my gratitude to my mentor, Dr. Cristina Hava Muntean, for her guidance and support with the research project. She has provided all the key details that are required to complete the project successfully.

References

Cai, L., Dong, J. and Wei, M. (2020). Multi-modal emotion recognition from speech and facial expression based on deep learning, *2020 Chinese Automation Congress (CAC)*, IEEE, pp. 5726–5729.

Chen, L., Ouyang, Y., Zeng, Y. and Li, Y. (2020). Dynamic facial expression recognition model based on bilstm-attention, *2020 15th International Conference on Computer Science & Education (ICCSE)*, IEEE, pp. 828–832.

Gill, R. and Singh, J. (2021). A deep learning approach for real time facial emotion recognition, *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*, IEEE, pp. 497–501.

Khaireddin, Y. and Chen, Z. (2021). Facial emotion recognition: State of the art performance on fer2013, *arXiv preprint arXiv:2105.03588* .

Khare, S. K. and Bajaj, V. (2020). Time–frequency representation and convolutional neural network-based emotion recognition, *IEEE transactions on neural networks and learning systems* **32**(7): 2901–2909.

Khasanov, D., Primqulov, O. et al. (2021). Gradient descent in machine learning, *2021 International Conference on Information Science and Communications Technologies (ICISCT)*, IEEE, pp. 1–3.

Kurien, R., Paila, A. R. and Nagendra, A. (2014). Application of paralysis analysis syndrome in customer decision making, *Procedia Economics and Finance* **11**: 323–334.

Liu, D., Bellotto, N. and Yue, S. (2019). Deep spiking neural network for video-based disguise face recognition based on dynamic facial movements, *IEEE transactions on neural networks and learning systems* **31**(6): 1843–1855.

MADUPU, R. K., KOTHAPALLI, C., YARRA, V., HARIKA, S. and BASHA, C. Z. (2020). Automatic human emotion recognition system using facial expressions with convolution neural network, *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, IEEE, pp. 1179–1183.

Munasinghe, M. (2018). Facial expression recognition using facial landmarks and random forest classifier, *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, IEEE, pp. 423–427.

Naas, S.-A. and Sigg, S. (2020). Real-time emotion recognition for sales, *2020 16th International Conference on Mobility, Sensing and Networking (MSN)*, IEEE, pp. 584–591.

Shrivastava, V., Richhariya, V. and Richhariya, V. (2018). Puzzling out emotions: A deep-learning approach to multimodal sentiment analysis, *2018 International Conference on Advanced Computation and Telecommunication (ICACAT)*, IEEE, pp. 1–6.

Yen, C.-T. and Li, K.-H. (2022). Discussions of different deep transfer learning models for emotion recognitions, *IEEE Access* **10**: 102860–102875.

Zhou, Y., Ren, F., Nishide, S. and Kang, X. (2019). Facial sentiment classification based on resnet-18 model, *2019 International Conference on electronic engineering and informatics (EEI)*, IEEE, pp. 463–466.