Bertran Maurelia Peidro

Research Paper

Abstract

The research question put forward by this paper is “To what extent can emotion detection artificial intelligence algorithms be used in order to improve education, business success, employee safety, healthcare and video game testing?” This paper looks to be able to give the consumers of the solution the ability to pinpoint which emotions educators, healthcare workers, video game testers and consumers of their business are feeling in order to improve their quality. In all of the areas that have been previously mentioned, there is a large inability to be able to identify when certain workers are feeling certain emotions, as a result of lack of recall memory, lack of honesty, or simply lack of awareness. In order to identify the satisfaction and dissatisfaction of healthcare workers, educators, video game testers and consumers of a business at certain moments, the quality of healthcare, education, video games and business websites will all increase respectively by being able to identify when workers are feeling a certain way, and how to reinforce or reduce the emotions being felt during those moments.

In order to solve this problem, many classification algorithms will be explored, such as convolutional neural networks, K nearest neighbor, multi-layer perceptron classifier, amongst many others. These classification algorithms will look for a successful and quality classification of the images being presented in order to ensure excellence whenever this solution is implemented in the areas mentioned previously.

By using the algorithms mentioned previously, the accuracy of the model was significantly higher than what it would have been had it been chosen randomly. For instance, with a convolutional neural network, there was an accuracy of 25%, whereas random assignment would have had a success rate of 14.29%

The conclusions from this investigation are that artificial intelligence algorithms can be used in order to classify images into 7 categories according to the emotion being displayed on them, and this solution can and will be successfully implemented into the areas mentioned previously.

Background

The paper titled *Facial expression recognition from video sequences: temporal and static modeling* (link) is based on how to identify facial expressions based on images and videos. The scientists that conducted these investigations got to the conclusion that the best models in order to classify these images were hidden Markov models (HMMs). Therefore, this research paper will benefit from this source by considering using these models in order to classify the images. Furthermore, the article titled *The True Failure Rate of Small Businesses* identified that by their first year, 20% of small businesses will have failed. By the second, 30% will have failed. After 5 years, half will have failed. By a decade, 70% of businesses will fail. Thus, the solution proposed by this project will allow this failure rate to decrease by allowing these businesses to identify when their products are being successful, and when they are not.

Problem framing

As mentioned in the introduction, the goal is to be able to create a tool in order to give a wide variety of groups the opportunity to correctly assess the emotion of their target, given their specific purpose.

In most problems, non-machine learning solutions are desired in order to lower the complexity of the problem, making it much easier to then distribute. In this situation, a very simple non-ML (machine learning) solution would involve handing questionnaires to the

target groups, in which they can assess their feelings towards the product or environment, as well as pointing to the moment in which said feelings arose. This would be an extremely simple method that could be implemented using pencil and paper.

Nonetheless, we have to take into consideration how fleeting human memory is. According to a study conducted on memory (citation), humans tend to forget extremely quickly and are often unable to recount events that occurred a few moments ago. Thus, asking participants to pinpoint the exact moment at which exact feelings arose can be quite difficult, and may give the recipient of the results a false result as to what is actually going erroneously with the product.

At the end of this paper, a comparison with a non-ML approach will be done, in order to compare the cost and feasibility of the two, and seeing which one is more appropriate. If there are only small improvements seen, all the cost and maintenance that goes into training an ML model are not justified, seeing as it would be better to spend those resources elsewhere, whilst still having a slightly less accurate model.

This particular problem will be a multiclass-single label classification model, in which the system will be able to classify an emotion from an image.

Generative AI is not a suitable solution for this problem, as this is a classification problem, in which the main goal is to be able to identify the face in a picture, not being able to generate one.

Dataset

The dataset being used is a free, open source collection of images from the website for machine learning and data science called Kaggle, uploaded by Jonathan Oheix. There are a total of 35,887 visual images in this data set. The data set came divided between a testing and a validation set, each one having 7 subfolders including thousands of images for the emotions being used: anger, disgust, fear, happiness, neutral, sadness, and surprise. All of these images

came with dimensions of 48x48 pixels and were in grayscale. The former feature will allow for a much more efficient neural network, as the input layer will be significantly smaller than if we used image dimensions from modern telephones. The latter will allow the convolutional neural network to not focus on factors such as color, illumination, and other factors in an image that may affect its classification. Further preprocessing steps that were implemented for this project included opening the images in the IDE using the PIL library, converting the images into a 1 dimensional array using the numpy library, as well as creating labels for each image type. Below are exemplar images from each dataset (these were expanded for visual purposes).


Angry images


Disgust image


Fear image


Happy image


Neutral image


Sad image


Surprise image

There are multiple criteria. that go into considering a dataset valuable for a machine learning approach, and this one fits all of these:

- Abundant: This dataset is abundant in images, having a total of 35,887 images. As there are more relevant and useful examples in the data set, it is better.

Methodology

For all of the data used in this investigation, the data was split into a training and testing set. These were created using the For Loops. After this procedure was carried out, four classifiers were used in order, and a final one using a mixture of the classifiers. For the first run of the classification model, 50 images were grabbed from each data set using the For Loop. Before each For Loop, the value of s was stated to be 0, thus, when the value of s became 50, the for loop stopped. After taking the first 50 images from each data set, the whole data set was used, and thus yielding accurate results. An important metric that was taken into consideration to assess the success of the modeling tasks was evaluating whether overfitting occurred. Overfitting is a process in which, after running an algorithm for a series

of epochs, it starts losing accuracy as a result of not giving appropriate weight to the different neurons in the testing accuracy. An easy way to notice whether a model has overfitted is by looking at the training and testing accuracy; if the former is significantly higher than the latter, then overfitting has most likely occurred.

The first model that was explored for this task was a multilayer perceptron classifier. At first, a test run used 50 iterations of each image to see the results. A multilayer perceptron classifier, also known as MLP, is a type of neural network which connects an input layer with an output layer using a variety of hidden layers. For this project, since all images are 48 by 48 pixels wide, there are 2,304 neurons in the input layer, and there are 7 neurons in the output layer, each representing a different class that can be obtained when classifying the images. These possible classifications are the ones that were mentioned previously, which are anger, disgust, fear, happiness, neutral, sadness, and surprise. During the preprocessing phase, the images are flattened, meaning that they are turned into a 1 dimensional array that allows the MLP to do calculations with it. All input neurons are connected to a hidden layer through lines that hold a weight. The value of the weight is then multiplied with the value from the neuron it is coming out of, and all of the values are then added up into a neuron. This process is repeated up to the output layer, which will give a probability of each of the 7 possible classifications. If the probability estimated by the MLP is the same as the actual class of the image (which was known to the neural network before starting), then the weights of the MLP are reinforced by increasing their values of the weights that would lead to a higher probability for the correct class. However, if the predictions were not correct, then the values of the weights are changed for the next iteration. For this project, the learning rate chosen was 0.0001, which represents the rate schedule for weight updates, meaning by how much the

weights are changed each time the model incorrectly classifies the images, and there were 2 layers with 100 neurons created.

The second classification model used for this task was K nearest neighbors. It is another type of algorithm for supervised learning. The way it works is that all data points are placed in a grid with assigned values for their classification. Once a new data point is inserted into the grid, the algorithm looks at the k nearest neighbors (k is one of the hyperparameters that can be changed when using this classifier) in order to predict what class the new point will belong to. The image below shows how the K nearest classifier algorithm was inserted into the code. For this project, the amount of k nearest neighbors used was 3, as seen in the parameters in the parenthesis.

The third classification model used for this task was Gaussian Process Classifier. This is another classifier used for supervised learning in machine learning. To understand this model, let us first describe what Gaussian probability is. This is a form of probability, in which, using a distribution curve, it is able to predict where a certain value will fall into based on the values from said distribution curve. The Gaussian Process Classifier is a step up in complexity from Gaussian probability. It can take into account much more factors, and is thus a very powerful regression algorithm.

The fourth classification model used for this task was SVC. This is another type of classification model used for supervised learning. The way that this algorithm works is that it randomly creates a function that divides the data points into two possible classifications. If this line does not separate the points into two, distinct classifications, then the process is repeated. The procedure is repeated until the right line with the classifications is obtained. In

the case of this project, since there are 7 possible classes, 6 lines are being created in order to split the data. The code below shows how SVC was implemented in this project's code.

The final classification model was a mixture of multiple models. At first, a multilayer perceptron classifier was used in order to classify images into one of the seven available classes. For every image that this classifier got wrong, these were added into a new dataset. Another multilayer perceptron classifier was then run on the images that were originally classified incorrectly, in order to increase the overall accuracy of the model.

Results and Discussion

MLP

For the first 50 images from each data set, the training accuracy was 100%, whilst the testing accuracy was only 15.6%, clearly showing that the model was overfitting. However, this is expected, as only 50 images were chosen from each data set.

Once all of the images were used to train the model, the training accuracy was 42.8% and the testing accuracy was 33.8%. Even though the results are significantly higher than if the images were randomly sorted, and they are higher than if the images were randomly classified, there is still room for improvement. Considering that this solution will eventually be sent to customers, it is of vital importance to ensure the highest possible accuracy.

K Nearest Neighbor

The results for only 50 images chosen were 56.2% for the training set, and 12.2% for the testing set. The results for the testing accuracy are quite mediocre, as they are below what would be obtained if the images were sorted randomly.

The results for using this model for the entire data set were 61.5% for the training set, and 32.1% for the testing set. These results are way better than the ones obtained with MLP

classification, however, considering the scope of this project, better results should be obtained.

Gaussian Classifier

The accuracy for the training set for only 50 images was 100%, whereas the one for the testing set was 15.6%. Clearly, the model is overfitting, as the accuracy for the training score is 100%, and less than 16% for the testing set. However, this is expected when working with such a limited data set.

No results were obtained with the Gaussian classifier when working with the entire data set, as it is too complex and powerful for this task, and it did not run after multiple hours.

SVC

The accuracy for the training set was 100%, whereas the one for the testing set was 11.1% when working with only 50 images from each class. The model is overfitting, as the training model accuracy is 100%, and the testing model is 11.1%, less than the accuracy obtained when randomly classifying the pictures. However, this is expected with such a small data sample.

The results for training the entire data set was 99.8%, and the results for the testing set was 30.7%. The model is clearly overfitting. However, this model had the highest accuracy considering both the training and testing accuracy.

Real world implications and ethical considerations

Representation

This algorithm takes into consideration that the only way for people to communicate is through facial expressions. Nonetheless, it completely ignores many groups that are unable to communicate through facial expression, such as those that suffer from autism, or any communication disorder. Thus, when considering how to improve the field in which this model is being applied, whether it be education or online businesses, it will leave out a huge proportion of individuals who are unable to communicate properly and will thus be unable to reap the rewards of these new technologies.

Transparency and consent

There is also the issue of receiving the consent from users to record their faces whilst watching them in order to. There are many issues nowadays with companies making use of cookies and selling their users' private information to advertising companies, with the most popular modern case being the Cambridge Analytica case in which Facebook was involved. If both legislations and users around the world feel uncomfortable and unsafe (rightly so) to divulge personal information such as their tendencies for "better ads", how will they feel about being monitored for "better products"? This raises serious concerns for this algorithm and those of a similar kind, especially if it were to fall into the hands of an oppressive regime wanting to control its citizens and measure their feelings at any point. What would happen if all those that had faces of "anger" or "boredom" during a dictator's speech were persecuted or punished? This product may give the rise to many opportunities for better products, but opens an even wider door to misuse and abuse of power.

Conclusion

In conclusion, this paper shows that machine learning algorithms can be used in order to classify emotions into different categories. The most promising results were from the

K-nearest neighbor model, with 61.5% accuracy in the training set, and 32.1% accuracy in the testing set. Although the results were successful, the more important implications are how this product will be used in the real world, and how effective it will be. On the one hand, if this product were to be used by the firms mentioned previously, although it could result in the intended purposes, it could also be used for more nefarious ones. On the one hand, facial expressions are not the only metric to assess emotions. Emotions are expressed through a variety of factors besides facial expressions, such as hormones, engagement, uniqueness, amongst others. Simplifying the emotions of individuals to their facial expression does not take into consideration how multifaceted and complex human feelings are. Furthermore, since firms will be using expressions as a metric to assess their success, some may force their users to show certain faces, which would result in biased results and a loss of authenticity.

On the more technical side, there are limitations for this model, which is the fact that it can only take in images with size 48 by 48 pixels, meaning that no other images can be classified. This is because of the amount of neurons that the algorithms have, which cannot be changed once the training set has been run.

In conclusion, although this project has successfully tackled the initial research question, there are many other aspects regarding its implementation that have not yet been considered, and their extent will never be known until its deployment.

Acknowledgements

I would like to thank my family for giving me the tools to carry out such wonderful and interesting projects that I am very passionate about.